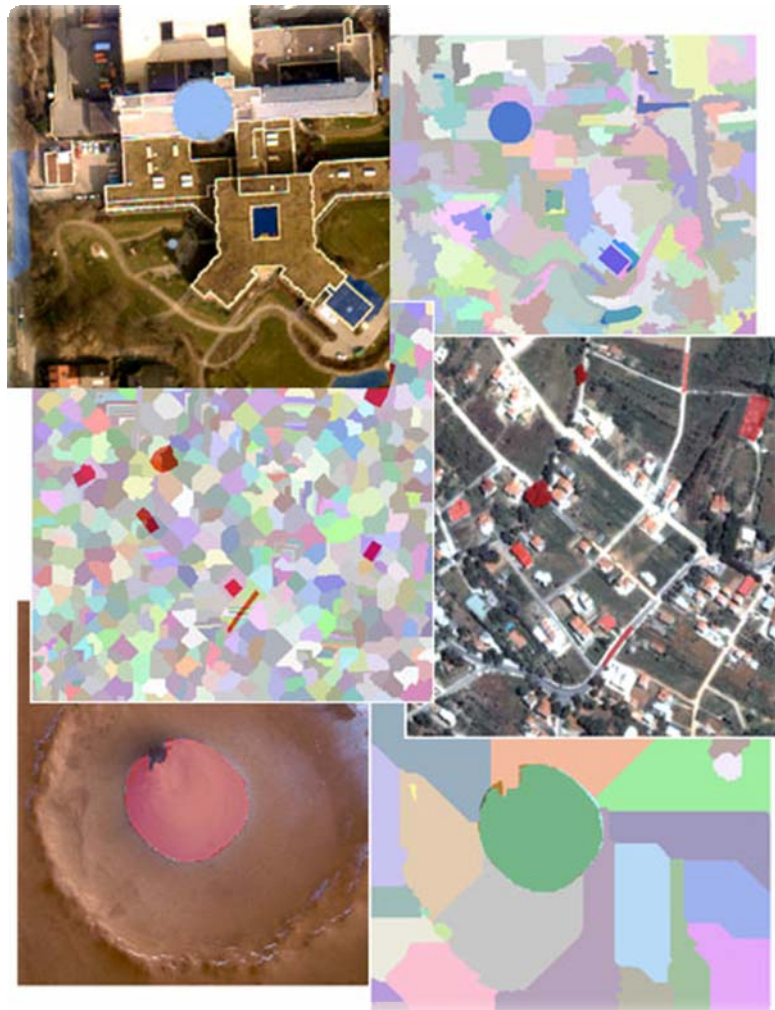




ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΑΓΡΟΝΟΜΩΝ ΚΑΙ ΤΟΠΟΓΡΑΦΩΝ ΜΗΧΑΝΙΚΩΝ
ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΓΕΩΠΛΗΡΟΦΟΡΙΚΗ
ΤΟΜΕΑΣ ΤΟΠΟΓΡΑΦΙΑΣ
ΕΡΓΑΣΤΗΡΙΟ ΤΗΛΕΠΙΣΚΟΠΗΣΗΣ

*«Αξιολόγηση Μεθόδων Κατάτμησης
Εικόνων Τηλεπισκόπησης, με Χρήση
Γενετικών Αλγορίθμων»*



ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
ΓΕΩΡΓΙΟΣ Λ. ΚΟΥΜΙΑΝΑΚΗΣ
ΙΟΥΛΙΟΣ 2013

Επιβλέπων: Αργιαλάς Δημήτριος, Καθηγητής ΕΜΠ

**...Στους γονείς μου και
στην οικογένεια μου...**

ΠΕΡΙΛΗΨΗ

Τα τελευταία χρόνια επικρατεί μια ραγδαία αύξηση του αριθμού των εικόνων που συναντάμε σε ψηφιακή μορφή και προέρχονται από διάφορες πηγές: ψηφιακές φωτογραφικές μηχανές, δορυφόροι, σαρωτές κτλ. Ο σκοπός της λήψης τους εξαρτάται από τον σκοπό για τον οποίο προορίζονται να χρησιμοποιηθούν. Ωστόσο κοινό παρανομαστή σε όλες αποτελεί η ανάγκη για ερμηνεία της περιεχόμενης πληροφορίας τους. Με τον όγκο των δεδομένων να αυξάνει συνεχώς και την υποκειμενικότητα της ανθρώπινης αντίληψης στην ερμηνεία τους, αναπτύσσεται έντονα η ανάγκη εύρεσης αυτόματων μεθόδων ερμηνείας των εικόνων.

Και ενώ η ερμηνεία των πληροφοριών που υπάρχουν σε μια εικόνα, μπορεί να γίνει με επιτυχία από τους φωτοερμηνευτές με σχετική ευκολία, δεν συμβαίνει το ίδιο με έναν εξελιγμένο ηλεκτρονικό υπολογιστή. Η σημασιολογική ανάλυση του περιεχομένου μιας εικόνας, μπορεί να αποτελεί μια προαιώνια βιολογική διεργασία του ανθρώπινου εγκεφάλου που εκτελείται συνεχώς, όμως για τον ηλεκτρονικό υπολογιστή, είναι μια πρόκληση των τελευταίων δεκαετιών.

Η αυτόματη εξαγωγή χαρακτηριστικών είναι ένα αντικείμενο ενδιαφέροντος πολλών πεδίων σύγχρονων εφαρμογών και βρίσκεται υπό συνεχή εξέταση και αναθεώρηση. Υπάρχει έντονη η ανάγκη αυτοματοποίησης της αναγνώρισης και χαρτογράφησης των τοπογραφικών, γεωμορφολογικών και ανθρωπογενών χαρακτηριστικών, ώστε η διαδικασία να καταστεί λιγότερο επίπονη χρονικά, χρησιμοποιώντας τις κατάλληλες μεθόδους αυτοματοποίησης.

Ωστόσο κατά καιρούς, έχουν αναπτυχθεί διάφοροι αλγόριθμοι, οι οποίοι έχουν σκοπό να βοηθήσουν τόσο στην φωτοερμηνεία όσο και στην κατάτμηση των εικόνων. Οι αλγόριθμοι αυτοί εμπλέκουν συνήθως πλήθος παραμέτρων, με αποτέλεσμα να καθίσταται ιδιαίτερα πολύπλοκη η χρησιμοποίησή τους και ακόμα δυσκολότερος ο εντοπισμός των βέλτιστων τιμών των παραμέτρων αυτών. Την διαδικασία αυτή έρχονται να απλουστεύσουν οι γενετικοί αλγόριθμοι μέσω των οποίων αναζητούνται οι βέλτιστες τιμές των παραπάνω παραμέτρων με βάση την χρήση μιας συνάρτησης αξιολόγησης, χωρίς ωστόσο να πραγματοποιείται δοκιμή όλων των τιμών των παραμέτρων αυτών.

Σκοπός της παρούσας διπλωματικής είναι η εφαρμογή γενετικών αλγορίθμων στην διαδικασία βελτιστοποίησης των παραμέτρων, οι οποίες χρησιμοποιούνται σε αλγορίθμους κατάτμησης εικόνων, υλοποιημένοι στην γλώσσα προγραμματισμού Python, με στόχο την ορθή κατάτμηση τηλεπισκοπικών και όχι μόνο απεικονίσεων, μέσω αυτοματοποιημένων διαδικασιών.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ:

Γενετικοί Αλγόριθμοι, Συνάρτηση Αξιολόγησης, Βελτιστοποίηση Παραμέτρων, Εργαλειοθήκη Γενετικών Αλγορίθμων, Γλώσσα Προγραμματισμού Python, Κατάτμηση Εικόνας, Αλγόριθμος Κατάτμησης, Δείκτης Κατάτμησης, Αντικείμενα Κατάτμησης, Τηλεπισκοπικές Απεικονίσεις, Δορυφορικές Εικόνες, Εικόνες Αναφοράς.

ABSTRACT

In recent years, there is a rapid increase in the number of images found in digital form and coming from several resources : digital cameras, satellites, scanners etc. The purpose of image collection depends on the purpose for which the images are intended to be used. However, the need for interpretation of the information contained in the images is a common denominator between all images. While the amount of data continues to increase and the subjectivity of the human perception continues to exist during their interpretation, a need for finding automatic methods of image interpretation is intensively being developed.

While the interpretation of the information contained in an image can be achieved quite easily by photo interpreters, it is not the same with an advanced computer. The semantic analysis of the image content may be a primordial biological process of the human brain that runs continuously, nevertheless it is a big challenge for the computer in the recent decades.

The automatic feature extraction is a subject of interest for several fields of modern applications and is under constant review and revision. There is a strong need for automating the identification and mapping of the topographic, geomorphologic and manmade features, so that the whole process can be less lengthy, using the proper automation methods.

However, various algorithms have occasionally been developed, which are designed to contribute not only in the photo-interpretation but also in the segmentation of images. These algorithms usually involve a grade number crowd of parameters, which makes their use very complicated and the identification of the best values of these parameters even harder. This process is being simplified by the genetic algorithms via which the best values of the parameters mentioned above are being sought through an evaluation function, without however testing all the values of those parameters.

The aim of this diploma thesis is the implementation of genetic algorithms for parameter optimization, of image segmentation algorithms implemented in the programming language Python, for the proper segmentation of remote sensing imagery.

KEY WORDS:

Genetic Algorithms, Fitness Function, Optimizing Parameters, Genetic Algorithm Toolbox Pyevolve, Programming Language Python, Image Segmentation, Segmentation Algorithm, Segmentation Indicator, Object of Segmentation, Remote Sensing Illustrations, Satellite Images, Reference Images.

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα μεταπτυχιακή εργασία είναι το αποτέλεσμα μιας σειράς αλληλεπιδράσεων με διάφορα άτομα, καθένα από τα οποία έπαιξε ένα σημαντικό ρόλο στην εξέλιξή της. Αξίζει, λοιπόν, να αφιερώσω την παρούσα σελίδα για να ευχαριστήσω ειλικρινά τα άτομα αυτά για την βοήθεια που μου προσέφεραν.

Την επίβλεψη της παρούσας μεταπτυχιακής εργασίας, που πραγματεύεται την διερεύνηση γενετικών αλγορίθμων σε αλγορίθμους κατάτμησης εικόνων, υλοποιημένους στην γλώσσα προγραμματισμού python, ανέλαβε ο καθηγητής του Ε.Μ.Π. κ. Αργιαλάς Δημήτριος, στον οποίο οφείλονται ευχαριστίες για την καθοδήγησή του, αλλά και για την εμπιστοσύνη που έδειξε στο πρόσωπό μου τόσο στην παρούσα εργασία όσο και στην διπλωματική εργασία της Σ.Α.Τ.Μ, στο να μου δώσει την ευκαιρία και δυνατότητα να επιλέξω ως μεταπτυχιακή εργασία ένα θέμα το οποίο με ενδιέφερε σημαντικά και με το οποίο είχα ασχοληθεί ήδη στην διπλωματική μου εργασία αλλά για το πρόβλημα της κατάτμησης μέσω μεθόδων ενεργών περιγραμμάτων.

Ευχαριστίες οφείλονται επίσης στον Υπ. Διδάκτορα κ. Τζώτσο Άγγελο, ο οποίος στάθηκε δίπλα μου όποτε τον χρειάστηκα, οποιαδήποτε ώρα και μέρα, βοηθώντας με σημαντικά με τις συμβουλές του και τις γνώσεις του στην πραγμάτωση της παρούσας εργασίας.

Δεν μπορώ να παραλείψω τις ευχαριστίες μου στην οικογένεια μου για την υπομονή που δείχνει όλα αυτά τα χρόνια που συνεχίζω τις σπουδές μου, στηρίζοντας με σε όλες μου τις αποφάσεις. Τέλος ευχαριστώ οποιονδήποτε άλλο φίλο, συμφοιτητή και συνάδελφο συνέβαλε τόσο στην πραγμάτωση της παρούσας εργασίας, όσο και στην ολοκλήρωση του μεταπτυχιακού προγράμματος, με την πολύτιμη συμβολή και συμβουλή του.

ΚΑΤΑΛΟΓΟΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1. ΕΙΣΑΓΩΓΗ	23
1.1 Γενικά	23
1.2 Ερμηνεία Τηλεπισκοπικών Απεικονίσεων	23
1.2.1 Οπτική προσέγγιση	23
1.2.2 Ψηφιακή προσέγγιση.....	25
1.3 Αδυναμίες	26
1.4 Απαιτούμενη Ανάγκη	26
1.5 Σκοπός της Διπλωματικής Εργασίας	27
1.6 Δομή της Διπλωματικής Εργασίας	27
2. ΑΝΑΣΚΟΠΗΣΗ ΒΙΒΛΙΟΓΡΑΦΙΑΣ	29
2.1 Γενικά	29
2.2 Γενετικοί Αλγόριθμοι	29
2.2.1 Γενικά	29
2.2.2 Ιστορική Αναδρομή	30
2.2.3 Ορολογία Γενετικών Αλγορίθμων	31
2.2.4 Κύρια Χαρακτηριστικά ενός Γενετικού Αλγορίθμου.....	32
2.2.5 Βασικά Στοιχεία ενός Γενετικού Αλγορίθμου.....	33
2.2.6 Πλεονεκτήματα Γενετικών Αλγορίθμων	41
2.2.7 Μειονεκτήματα Γενετικών Αλγορίθμων	42
2.3 Παρουσίαση της Βιβλιοθήκης Γενετικών Αλγορίθμων Pyevolve στην Γλώσσα Προγραμματισμού Python	43
2.3.1 Γενικά	43
2.3.2 Οργάνωση ενός Προβλήματος	44
2.3.3 Αναπαραστάσεις ενός Χρωμοσώματος	45
2.3.4 Ανάλυση της επιλογής Initializers	46
2.3.5 Ανάλυση της Επιλογής Population Size	46
2.3.6 Ανάλυση της Επιλογής Selection Methods.....	47
2.3.7 Ανάλυση της Επιλογής setElitism και setElitism Replacement.....	47
2.3.8 Ανάλυση της Επιλογής Crossover Methods	48
2.3.9 Ανάλυση της Επιλογής Crossover Rate	50
2.3.10 Ανάλυση της Επιλογής Mutator Methods	52
2.3.11 Ανάλυση της Επιλογής Mutation Rate	53
2.3.12 Ανάλυση των Βασικών Επιλογών Migration	53

2.3.12.1 MigrationSelector	53
2.3.12.2 MigrationRate.....	54
2.3.12.3 NumIndividuals	54
2.3.13 Ανάλυση των Επιλογών - Συνθηκών Τερματισμού του ΓΑ.....	54
2.3.13.1 Ανάλυση της επιλογής Generations	54
2.3.13.2 Ανάλυση της επιλογής ConvergeCriteria	54
2.3.13.3 Ανάλυση της επιλογής FitnessStatsCriteria	54
2.3.14 Ανάλυση της Επιλογής Fitness Scaling Methods	55
2.3.15 Ανάλυση της Επιλογής DBAdapters	55
2.3.16 Ανάλυση της Επιλογής Graphical Plotting Tool Options.....	55
2.4 Ανάλυση των Αλγορίθμων Κατάτμησης.....	55
2.4.1 Γενικά	55
2.4.2 Αλγόριθμος βασισμένος στον μετασχηματισμό του Υδροκρίτη (Watershed).....	56
2.4.3 Αλγόριθμος βασισμένος σε τεχνικές τοπικής ανάλυσης επέκτασης περιοχών (region growing)	58
2.4.4 Αλγόριθμος που βασίζεται σε τεχνικές τοπική ένωσης περιοχών (region merging).....	60
2.5 Προηγούμενες Μελέτες.....	62
3. ΜΕΘΟΔΟΛΟΓΙΑ.....	67
3.1 Γενικά	67
3.2 Ανάλυση του Υλοποιημένου Αλγορίθμου Ελέγχου Ακρίβειας της Κατάτμησης καθώς και του Γενετικού Αλγορίθμου	67
3.2.1 Γενικά	67
3.2.2 Ανάλυση του αλγορίθμου ελέγχου της ακρίβειας της κατάτμησης βασισμένη.....	67
3.2.3 Ανάλυση του γενετικού αλγορίθμου.....	79
3.3 Χρησιμοποιηθείσες Εικόνες.....	80
3.4 Δημιουργία Εικόνων Αναφοράς.....	83
3.5 Ανάλυση Ευαισθησίας – Επιλογή Παραμέτρων.....	88
3.5.1 Γενικά	88
3.5.2 Υλοποίηση Γενετικού Αλγορίθμου, με Χρησιμοποίηση του Αλγορίθμου Κατάτμησης Watershed	90
3.5.2.1 Υλοποίηση στην ΔΕ του Αγίου Στεφάνου Αττικής.....	90

3.5.2.2	Υλοποίηση στην εικόνα από αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας	99
3.5.2.3	Υλοποίηση στην ΔΕ της NASA για τον πλανήτη Άρη.....	108
3.5.3	Υλοποίηση Γενετικού Αλγορίθμου, με Χρησιμοποίηση του Αλγορίθμου Κατάτμησης Region Growing.....	115
3.5.3.1	Υλοποίηση στην ΔΕ του Αγίου Στεφάνου Αττικής	115
3.5.3.2	Υλοποίηση στην εικόνα από αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας	123
3.5.3.3	Υλοποίηση στην ΔΕ της NASA για τον πλανήτη Άρη.....	131
3.5.4	Υλοποίηση Γενετικού Αλγορίθμου, με Χρησιμοποίηση του Αλγορίθμου Κατάτμησης Region Merging	137
3.5.4.1	Υλοποίηση στην ΔΕ του Αγίου Στεφάνου Αττικής	137
3.5.4.2	Υλοποίηση στην εικόνα από αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας	143
3.5.4.3	Υλοποίηση στην ΔΕ της NASA για τον πλανήτη Άρη.....	149
4.	ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΡΟΤΑΣΕΙΣ	155
	ΒΙΒΛΙΟΓΡΑΦΙΚΕΣ ΠΗΓΕΣ	159
	ΠΑΡΑΡΤΗΜΑΤΑ.....	163

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 2.1-	Υπολογισμός απόδοσης για τον αρχικό πληθυσμό.....	36
Πίνακας 2.2 -	Πίνακας γειτόνων κάθε γονιδίου στο χρωμόσωμα.....	49
Πίνακας 2.3 -	Διαδικασία υλοποίησης της μεθόδου διασταύρωσης γειτνίασης.....	50
Πίνακας 3.1-	Στοιχεία εφαρμογής ΓΑ στην ΔΕ IKONOS περιοχής του Αγίου Στεφάνου Αττικής, με αλγόριθμο κατάτμησης watershed	91
Πίνακας 3.2 -	Στοιχεία εφαρμογής ΓΑ στην εικόνα από αερομεταφερόμενο σαρωτή σε περιοχή της Γερμανίας, με αλγόριθμο κατάτμησης watershed	100
Πίνακας 3.3 -	Στοιχεία εφαρμογής ΓΑ στην ΔΕ της NASA για τον πλανήτη Άρη, με αλγόριθμο κατάτμησης watershed.....	109
Πίνακας 3.4 -	Στοιχεία εφαρμογής ΓΑ στην ΔΕ IKONOS περιοχής του Αγίου Στεφάνου Αττικής, με αλγόριθμο κατάτμησης region growing	116
Πίνακας 3.5 -	Στοιχεία εφαρμογής ΓΑ στην εικόνα από αερομεταφερόμενο σαρωτή σε περιοχή της Γερμανίας, με αλγόριθμο κατάτμησης region growing	124
Πίνακας 3.6 -	Στοιχεία εφαρμογής ΓΑ στην ΔΕ της NASA για τον πλανήτη Άρη, με αλγόριθμο κατάτμησης region growing.....	132

Πίνακας 3.7 - Στοιχεία εφαρμογής ΓΑ στην ΔΕ IKONOS περιοχής του Αγίου Στεφάνου Αττικής με αλγόριθμο κατάτμησης region merging	138
Πίνακας 3.8 - Στοιχεία εφαρμογής ΓΑ στην εικόνα από αερομεταφερόμενο σαρωτή σε περιοχή της Γερμανίας, με αλγόριθμο κατάτμησης region merging	144
Πίνακας 3.9 - Στοιχεία εφαρμογής ΓΑ στην ΔΕ της NASA για τον πλανήτη Άρη, με αλγόριθμο κατάτμησης region merging	150

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 2.1- Σχηματική αναπαράσταση εξαναγκασμένης ρουλέτας	37
Εικόνα 2.2 - Διάγραμμα ροής της εφαρμογής του ΓΑ.....	40
Εικόνα 2.3 - Διασταύρωση μονού σημείου	48
Εικόνα 2.4 - Διασταύρωση διπλού σημείου	48
Εικόνα 2.5 - Διασταύρωση ομοιόμορφη	49
Εικόνα 2.6 - Διάγραμμα καλύτερης και μέσης τιμής της συνάρτησης αξιολόγησης με τιμή Crossover Rate ίση με 0.8	51
Εικόνα 2.7 - Διάγραμμα καλύτερης και μέσης τιμής της συνάρτησης αξιολόγησης με τιμή Crossover Rate ίση με 1	51
Εικόνα 2.8 - Διάγραμμα καλύτερης και μέσης τιμής της συνάρτησης αξιολόγησης με τιμή Crossover Rate ίση με 0	52
Εικόνα 2.9 - Αρχική εικόνα	57
Εικόνα 2.10 - Εφαρμογή φίλτρου Gauss 3 x 3	57
Εικόνα 2.11 - Εφαρμογή φίλτρου Gauss 5 x 5	57
Εικόνα 2.12 - Εφαρμογή φίλτρου Gauss 7 x 7	57
Εικόνα 2.13 - Αποτέλεσμα κατάτμησης με χρησιμοποίηση εικόνας μορφής .tif ...	60
Εικόνα 2.14 - Αποτέλεσμα κατάτμησης με χρησιμοποίηση εικόνας μορφής .png	60
Εικόνα 3.1 - Γεωμετρικές σχέσεις μεταξύ πολυγώνων αναφοράς και πολυγώνων κατάτμησης.....	68
Εικόνα 3.2 - Εικόνα από αερομεταφερόμενη κάμερα DMC σε περιοχή της Γερμανίας με διαστάσεις 200 x 300 pixel.....	81
Εικόνα 3.3 - Εικόνα από αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας με διαστάσεις 473 x 474 pixel.....	81
Εικόνα 3.4 - ΔΕ Ιkonos της περιοχής Αγ. Στεφάνου Αττικής με διαστάσεις 501 x 501 pixel.....	82

Εικόνα 3.5 - ΔΕ NASA του πλανήτη Άρη με διαστάσεις 299 x 307 pixel	82
Εικόνα 3.6 - Γραφική απεικόνιση του εργαλείου Vect_to_Raster μέσω του Model Builder του λογισμικού ESRI ArcGIS 10.....	83
Εικόνα 3.7 - Μορφή πίνακα ιδιοτήτων πολυγωνικού αρχείου για την δημιουργία της τελικής εικόνας αναφοράς	83
Εικόνα 3.8 - Εμφάνιση παραθύρου από την εφαρμογή του εργαλείου Vect_to_Raster.....	84
Εικόνα 3.9 - Μορφή πίνακα ιδιοτήτων της τελικής εικόνας αναφοράς	85
Εικόνα 3.10 - Εικόνα αναφοράς από την εφαρμογή του εργαλείου Vect_to_Raster στην εικόνα 3.2 (Περιοχή calw Γερμανίας, εταιρία Intergraph)	85
Εικόνα 3.11 - Εικόνα αναφοράς από την εφαρμογή του εργαλείου Vect_to_Raster στην εικόνα 3.3 (Περιοχή Γερμανίας εταιρία Toposys)	86
Εικόνα 3.12 - Εικόνα αναφοράς από την εφαρμογή του εργαλείου Vect_to_Raster στην εικόνα 3.4 (Περιοχή Αγ. Στεφάνου Αττικής).....	86
Εικόνα 3.13 - Εικόνα αναφοράς από την εφαρμογή του εργαλείου Vect_to_Raster στην εικόνα 3.5 (Πλανήτη Άρης από δορυφόρο της NASA)	87
Εικόνα 3.14 - Διάγραμμα ροής εφαρμογής του ΓΑ με βάση τους αλγορίθμους κατάτμησης και τις χρησιμοποιηθέντες εικόνες	89
Εικόνα 3.15 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα του Αγ. Στεφάνου	92
Εικόνα 3.16 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED2 (watershed - Min Fitness / Min Raw = 0,82 / 0,70).....	93
Εικόνα 3.17 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	93
Εικόνα 3.18 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	93
Εικόνα 3.19 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	94
Εικόνα 3.20 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	94
Εικόνα 3.21 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	94
Εικόνα 3.22 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	94

Εικόνα 3.23 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED1 (watershed - Min Fitness / Min Raw = 0,28 / 0,11).....	95
Εικόνα 3.24 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	95
Εικόνα 3.25 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	95
Εικόνα 3.26 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED1 (watershed - Min Fitness / Min Raw = 0,23 / 0,26).....	96
Εικόνα 3.27 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	96
Εικόνα 3.28 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	96
Εικόνα 3.29 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη $(ED1+ED2)/2$ (watershed - Min Fitness / Min Raw = 1,14 / 0,60).....	97
Εικόνα 3.30 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	98
Εικόνα 3.31 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	98
Εικόνα 3.32 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης με δείκτη ED2.....	98
Εικόνα 3.33 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης με δείκτη $(ED1+ED2)/2$	98
Εικόνα 3.34 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα σε περιοχή της Γερμανίας	101
Εικόνα 3.35 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED2 (watershed - Min Fitness / Min Raw = 1,5 / 0,43).....	102
Εικόνα 3.36 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED2 (watershed - Min Fitness / Min Raw = 0,0 / 0,77).....	103
Εικόνα 3.37 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 1,5 / 0,43 - δεξιά Min Fitness / Min Raw = 0,0 / 0,77).....	103

Εικόνα 3.38 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 1,5 / 0,43 - δεξιά Min Fitness / Min Raw = 0,0 / 0,77).....	104
Εικόνα 3.39 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 1,5 / 0,43 - δεξιά Min Fitness / Min Raw = 0,0 / 0,77).....	104
Εικόνα 3.40 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 1,5 / 0,43 - δεξιά Min Fitness / Min Raw = 0,0 / 0,77).....	104
Εικόνα 3.41 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED1 (watershed - Min Fitness / Min Raw = 0,22 / 0,19).....	105
Εικόνα 3.42 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	106
Εικόνα 3.43 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	106
Εικόνα 3.44 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη (ED1+ED2)/2 (watershed - Min Fitness / Min Raw = 1,14 / 0,60).....	107
Εικόνα 3.45 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	107
Εικόνα 3.46 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	107
Εικόνα 3.47 - Αντικείμενο αναφοράς σε επικάλυψη με την εικόνα του πλανήτη Άρη	110
Εικόνα 3.48 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (watershed - Min Fitness / Min Raw = 0,97 / 0,0).....	111
Εικόνα 3.49 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (watershed - Min Fitness / Min Raw = 1,54 / 0,0).....	111
Εικόνα 3.50 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED1 (watershed - Min Fitness / Min Raw = 0,11 / 0,02).....	112

Εικόνα 3.51 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη $(ED1+ED2)/2$ (watershed - Min Fitness / Min Raw = 0,32/0,017).....	113
Εικόνα 3.52 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη $(ED1+ED2)/2$ (watershed - Min Fitness / Min Raw = 0,38 / 0,02).....	114
Εικόνα 3.53 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα του Αγ. Στεφάνου	117
Εικόνα 3.54 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED2 (Region Growing - Min Fitness / Min Raw = 126 / 101).....	118
Εικόνα 3.55 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	118
Εικόνα 3.56 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	118
Εικόνα 3.57- Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	119
Εικόνα 3.58 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	119
Εικόνα 3.59 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED1 (Region Growing - Min Fitness / Min Raw = 0,43 / 0,70).....	120
Εικόνα 3.60 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	120
Εικόνα 3.61 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	120
Εικόνα 3.62 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	121
Εικόνα 3.63 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	121
Εικόνα 3.64 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη $(ED1+ED2)/2$ (Region Growing - Min Fitness / Min Raw = 67 / 51)	122
Εικόνα 3.65 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	122

Εικόνα 3.66 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	122
Εικόνα 3.67 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα σε περιοχή της Γερμανίας.....	125
Εικόνα 3.68 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED2 (Region Growing - Min Fitness / Min Raw = 131/42,86).....	126
Εικόνα 3.69 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED2 (Region Growing - Min Fitness / Min Raw = 42,1 / 49,2).....	127
Εικόνα 3.70 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 131 / 42,86) - δεξιά Min Fitness / Min Raw = 42,1 / 49,2).....	127
Εικόνα 3.71 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 131 / 42,86) - δεξιά Min Fitness / Min Raw = 42,1 / 49,2).....	128
Εικόνα 3.72 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 131 / 42,86) - δεξιά Min Fitness / Min Raw = 42,1 / 49,2).....	128
Εικόνα 3.73 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED1 (Region Growing - Min Fitness / Min Raw = 0,61 / 0,70).....	129
Εικόνα 3.74 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	129
Εικόνα 3.75 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	129
Εικόνα 3.76 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη $(ED1+ED2)/2$ (Region Growing - Min Fitness / Min Raw = 22 / 21)	130
Εικόνα 3.77 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	131
Εικόνα 3.78 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	131
Εικόνα 3.79 - Αντικείμενο αναφοράς σε επικάλυψη με την εικόνα του πλανήτη Άρη	133

Εικόνα 3.80 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (Region Growing - Min Fitness / Min Raw = 1,27 / 0,158).....	134
Εικόνα 3.81 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (Region Growing - Min Fitness / Min Raw = 3,44 / 0,158).....	134
Εικόνα 3.82 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED1 (Region Growing - Min Fitness / Min Raw = 0,20 / 0,02).....	135
Εικόνα 3.83 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη (ED1+ED2)/2 (Region Growing - Min Fitness / Min Raw = 0,91 / 0,13)	136
Εικόνα 3.84 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα του Αγ. Στεφάνου	139
Εικόνα 3.85 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED2 (Region Merging - Min Fitness / Min Raw = 17,19 / 8,96).....	140
Εικόνα 3.86 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	140
Εικόνα 3.87 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	140
Εικόνα 3.88 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED1 (Region Merging - Min Fitness / Min Raw = 0,28 / 0,66).....	141
Εικόνα 3.89 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	141
Εικόνα 3.90 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	141
Εικόνα 3.91 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη (ED1+ED2)/2 (Region Merging - Min Fitness / Min Raw = 7,73/4,08).....	142
Εικόνα 3.92 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	143
Εικόνα 3.93 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	143

Εικόνα 3.94 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα σε περιοχή της Γερμανίας	145
Εικόνα 3.95 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED2 (Region Merging - Min Fitness / Min Raw = 1,39 / 0,85).....	146
Εικόνα 3.96 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	146
Εικόνα 3.97 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	146
Εικόνα 3.98 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED1 (Region Merging - Min Fitness / Min Raw = 0,39 / 0,35).....	147
Εικόνα 3.99 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	148
Εικόνα 3.100 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης.....	148
Εικόνα 3.101 - Αντικείμενα κατάτμησης με βάση την χρήση του δείκτη ED2 ως συνάρτηση αξιολόγησης (region merging)	148
Εικόνα 3.102 - Αντικείμενα κατάτμησης με βάση την χρήση του δείκτη ED2 ως συνάρτηση αξιολόγησης (region merging)	148
Εικόνα 3.103 - Αντικείμενο αναφοράς σε επικάλυψη με την εικόνα του πλανήτη Άρη.....	151
Εικόνα 3.104 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (Region Merging - Min Fitness / Min Raw = 0 / 0,04).....	152
Εικόνα 3.105 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (Region Merging - Min Fitness / Min Raw = 2,70 / 0,02).....	152
Εικόνα 3.106 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED1 (Region Merging - Min Fitness / Min Raw = 0,005 / 0,0012).....	153
Εικόνα 3.107 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη (ED1+ED2)/2 (Region Merging - Min Fitness / Min Raw = 0,88 / 0,07).....	154

1. ΕΙΣΑΓΩΓΗ

1.1 Γενικά

Τα τελευταία χρόνια η παρακολούθηση της γης μέσω δορυφόρων από το διάστημα έχει συμβάλει στην επίλυση αλλά και στην πρόληψη μεγάλου αριθμού προβλημάτων τόσο μέσω των δράσεων αλλά και μέσω του σχεδιασμού αναπτυξιακών και παραγωγικών δραστηριοτήτων. Η αξιοποίηση τηλεπισκοπικών απεικονίσεων για τη μελέτη της γης άρχισε τη δεκαετία του 1960, όταν οι τεχνικές δυνατότητες των δορυφόρων που κινούνταν σε τροχιά γύρω από τη γη, ακολούθησαν τη ραγδαία ανάπτυξη των δυνατοτήτων των Η/Υ για τη διαχείριση μεγάλου όγκου δεδομένων.

Οι τηλεπισκοπικές απεικονίσεις είναι καταγραφές της ανακλώμενης ή/και της εκπνευμένης, οπισθοσκεδαζόμενης ηλεκτρομαγνητικής ακτινοβολίας. Προκειμένου, λοιπόν, να γίνει δυνατή η συλλογή χρήσιμης πληροφορίας, είναι απαραίτητη η ερμηνεία τους.

1.2 Ερμηνεία Τηλεπισκοπικών Απεικονίσεων

Η ερμηνεία των τηλεπισκοπικών απεικονίσεων μπορεί να γίνει είτε οπτικά είτε ψηφιακά (αυτόματα). Η οπτική ερμηνεία στηρίζεται κυρίως στην ποιοτική φωτοερμηνεία των εικόνων ενώ η ψηφιακή στηρίζεται στην ποσοτική κατάταξη των εικονοστοιχείων μιας εικόνας σε συγκεκριμένες φασματικές κατηγορίες ή σε μεθόδους αναγνώρισης και εξαγωγής αντικειμένων. Τόσο η ποιοτική όσο και η ποσοτική μέθοδος έχουν και πλεονεκτήματα και μειονεκτήματα.

Η επιτυχία της φωτοερμηνείας εξαρτάται από την επιδεξιότητα και εμπειρία του φωτοερμηνευτή να χρησιμοποιήσει αποτελεσματικά τις χωρικές, φασματικές και διαχρονικές πληροφορίες που αφορούν συγκριμένο γνωστικό πεδίο.

1.2.1 Οπτική προσέγγιση

Η διαδικασία της οπτικής αναγνώρισης η οποία συνδέεται με την ποιοτική προσέγγιση για ερμηνεία αποτελείται από τα εξής βήματα:

- ✓ **Διαδικασία ανίχνευσης:** Η ανίχνευση εξαρτάται από τη διακριτική ικανότητα του δέκτη, η οποία ποικίλλει ανάλογα με την εφαρμογή και το είδος του δορυφόρου. Η διακριτική ικανότητα διακρίνεται σε:
 - Χωρική διακριτική ικανότητα (Spatial Resolution). Είναι η ικανότητα του δέκτη να ξεχωρίζει πολύ κοντινά αντικείμενα ή πληροφορίες στην απεικόνιση (ποσοτικά ισούται με τη μικρότερη απόσταση που μπορούν να έχουν δύο αντικείμενα, έτσι ώστε να εμφανίζονται ξεχωριστά),
 - Χρονική διακριτική ικανότητα (Temporal Resolution), η οποία σχετίζεται με τη συχνότητα λήψης απεικονίσεων για την ίδια γεωγραφική περιοχή,
 - Ραδιομετρική διακριτική ικανότητα (Radiometric Resolution), που ορίζεται ως η ευαισθησία του δέκτη να καταγράφει διαφορές στην ισχύ του σήματος,
 - Φασματική διακριτική ικανότητα (Spectral Resolution), η οποία περιλαμβάνει το εύρος των περιοχών του φάσματος της ηλεκτρομαγνητικής ακτινοβολίας, στις

οποίες πραγματοποιεί καταγραφές ένας πολυφασματικός δέκτης, καθώς και τον αριθμό των καναλιών που χρησιμοποιούνται.

- ✓ **Διαδικασία αναγνώρισης** αντικειμένων/εμφανίσεων στην περιοχή ενδιαφέροντος. Η αναγνώριση περιλαμβάνει τα εξής στάδια:
 - στάδιο αναγνώρισης γενικών κατηγοριών και σχημάτων,
 - στάδιο ανάλυσης ομοιοτήτων και διαφορών και τέλος,
 - στάδιο της συσχέτισης με το περιβάλλον, όπου χρησιμοποιούνται φωτοερμηνευτικά κλειδιά, πορίσματα επιγείων ελέγχων και δειγματοληψιών. Τα βασικά φωτοαναγνωριστικά στοιχεία που χρησιμοποιούνται είναι: ο τόνος, το χρώμα, το μέγεθος, το σχήμα, το πρότυπο, η υφή, η σκιά, η θέση / τοποθεσία και η σχέση με το περιβάλλον.
- ✓ **Διαδικασία ανάλυσης** η οποία περιλαμβάνει την οριοθέτηση των περιοχών που παρουσιάζουν ομοιογένεια ως προς τα χαρακτηριστικά τους.
- ✓ **Διαδικασία οριοθέτησης** σε κατηγορίες η οποία και οδηγεί στην δημιουργία ενός θεματικού χάρτη της περιοχής που μας ενδιαφέρει, ο οποίος με τη σειρά του, μπορεί να ενσωματωθεί σε ένα Γεωγραφικό Σύστημα Πληροφοριών (G.I.S).

Τα πλεονεκτήματα της προσέγγισης αυτής (Αργιαλάς, 1998) χαρακτηρίζονται τα εξής:

- ✓ Άμεση επίδραση του ανθρώπινου παράγοντα με αποτέλεσμα υψηλού επιπέδου νοητικές αποφάσεις και συμπεράσματα,
- ✓ Παρουσιάζει βέλτιστα αποτελέσματα σε χωρικές εκτιμήσεις,
- ✓ Η αναγνώριση του σχήματος, των προτύπων υφής και των αντικειμένων είναι σχετικά εύκολη, για έμπειρους φωτοερμηνευτές σε εξειδικευμένα αντικείμενα,
- ✓ Όταν υπάρχουν έμπειροι φωτοερμηνευτές εξάγονται εύκολα συμπεράσματα,
- ✓ Οι γνώσεις του χρήστη μπορούν να χρησιμοποιηθούν κατά ποιοτικό τρόπο.

Από την άλλη πλευρά τα μειονεκτήματα της προσέγγισης είναι τα εξής:

- ✓ Δεν είναι αποτελεσματική η φωτοερμηνεία στον ακριβή καθορισμό της έκτασης των επιφανειών διαφόρων ειδών εδαφοκάλυψης,
- ✓ Μόνο τρία κανάλια μπορούν να εξεταστούν ταυτόχρονα από τον φωτοερμηνευτή με αποτέλεσμα να περιορίζεται η φωτοερμηνευτική διαδικασία,
- ✓ Ο φωτοερμηνευτής δεν είναι σε θέση να χρησιμοποιήσει όλο το εύρος της διαθέσιμης ραδιομετρικής διαχωριστικής ικανότητας των περισσότερων σαρωτών,
- ✓ Ο φωτοερμηνευτής μπορεί να αφομοιώσει ελάχιστα μόνο εικονοστοιχεία σε μία δεδομένη χρονική στιγμή,

- ✓ Γενικά είναι ιδιαίτερα **χρονοβόρα**, με αποτέλεσμα να είναι αδύνατη η επεξεργασία μεγάλης ποσότητας δεδομένων.

1.2.2 Ψηφιακή προσέγγιση

Η διαδικασία της ψηφιακής αναγνώρισης, για την οποία και ευθύνεται κυρίως ο ηλεκτρονικός υπολογιστής, αποτελεί ίσως τη μοναδική λύση η οποία συνδέεται με την ποσοτική προσέγγιση για ερμηνεία, αποτελείται δε από τα εξής βήματα:

- ✓ Μετατροπή της αναλογικής απεικόνισης σε ψηφιακή μορφή, αν πρόκειται για αεροφωτογραφίες (το βήμα αυτό δεν χρειάζεται, αν πρόκειται για δεδομένα που προέρχονται από δορυφόρους και λαμβάνονται απευθείας σε ψηφιακή μορφή),
- ✓ Επεξεργασία δεδομένων, που περιλαμβάνει συνήθως ραδιομετρική και γεωμετρική διόρθωση,
- ✓ Εξαγωγή χαρακτηριστικών (feature extraction). Στο στάδιο αυτό επιλέγονται οι μετρήσεις που είναι απαραίτητες για την αναγνώριση ενός αντικείμενου/εμφάνισης και την **ταξινόμησή** του. Τα χαρακτηριστικά μπορεί να είναι χωρικά, φασματικά και χρονικά.

Προκειμένου να πραγματοποιηθεί η ταξινόμηση εφαρμόζεται μια από τις παρακάτω δυο μεθόδους (Αργιαλάς, 1998):

- ☞ **Επιβλεπόμενη ταξινόμηση:** Απαιτεί τη συλλογή από τον χρήστη δεδομένων εκπαίδευσης από κάθε θεματική κατηγορία. Το σύνολο των δεδομένων εκπαίδευσης συνίσταται από εικονοστοιχεία με γνωστή εκ των προτέρων ταυτότητα που ομαδοποιεί ο χρήστης, συνήθως σε πολύγωνα δεδομένων εκπαίδευσης πάνω στην οθόνη του συστήματος επεξεργασίας εικόνων και που χρησιμοποιούνται για την επίβλεψη του αλγορίθμου ταξινόμησης. Τα δεδομένα αυτά, τα οποία δεν είναι παρά μια ομάδα pixels, θα “εκπαιδεύσουν” τον υπολογιστή να αναγνωρίζει τα διάφορα αντικείμενα/εμφάνισης (θεματικές τάξεις), μέσω προσδιορισμού στατιστικών παραμέτρων, όπως μέσους όρους, τυπική απόκλιση, πίνακες μεταβλητότητας - συμμεταβλητότητας. Στη συνέχεια οι παράμετροι αυτές συγκρίνονται με τις αντίστοιχες παραμέτρους άλλων περιοχών της απεικόνισης, που είναι άγνωστες, μέσω συγκεκριμένου κάθε φορά αλγορίθμου ταξινόμησης, όπως ο αλγόριθμος μεγίστης πιθανοφάνειας, ελάχιστης απόστασης, παραλληλεπιπέδου, Mahalanobis κλπ, με τελικό αποτέλεσμα την ταξινόμηση όλων των pixels της απεικόνισης σε θεματικές τάξεις.
- ☞ **Μη επιβλεπόμενη ταξινόμηση:** Στη μέθοδο αυτή σε αντίθεση με την προηγούμενη δεν χρησιμοποιούνται δεδομένα εκπαίδευσης. Χωρίς δηλαδή να πραγματοποιείται “εκπαίδευση” του υπολογιστή η ψηφιακή πολυφασματική εικόνα κατηγοριοποιείται σε ομοιογενείς φασματικές τάξεις, ανάλογα με τη “φασματική τους υπογραφή”, αλλά όχι σε θεματικές. Η σχέση των ταξινομηθέντων φασματικών τάξεων με τις επιθυμητές θεματικές τάξεις προσδιορίζεται μετά το στάδιο της ταξινόμησης με τη βοήθεια εργασίας πεδίου, με επίγειους ελέγχους, με σύγκριση αεροφωτογραφιών. Η μη επιβλεπόμενη προσέγγιση περιλαμβάνει τις μεθόδους ανάλυσης συσσωρεύσεων, τις μεθόδους επιλογής επικρατουσών τιμών του ιστογράμματος κ.α.

Ωστόσο στην ψηφιακή προσέγγιση ανήκει και η μεθοδολογία της εξαγωγής χαρακτηριστικών - αντικειμένων από μια εικόνα (κατάτμηση). Η μέθοδος αυτή διακρίνεται σε δυο κατηγορίες. Η πρώτη βασίζεται στην εύρεση ορίων ενώ η δεύτερη στην εύρεση περιοχών. Η εύρεση ορίων αναφέρεται σε τεχνικές ανίχνευσης ακμών και εξέλιξης ενεργών περιγραμμάτων, ενώ η εύρεση περιοχών βασίζεται σε τεχνικές τοπικής ανάλυσης επέκτασης - ένωσης περιοχών, σε τεχνικές στατιστικής ανάλυσης, στην θεωρία των γράφων και σε μεταβολικές μεθόδους.

Τα πλεονεκτήματα της προσέγγισης αυτής χαρακτηρίζονται τα εξής:

- ✓ Απαιτεί μικρή ανθρώπινη παρέμβαση και συμβολή,
- ✓ Διαθέτει μεγάλη ποσοτική ακρίβεια η οποία προέρχεται από την δυνατότητα του υπολογιστή να επεξεργαστεί το κάθε εικονοστοιχείο σε μια εικόνα,
- ✓ Μπορεί να χρησιμοποιηθεί για να επεξεργαστεί μια εικόνα στο επίπεδο του εικονοστοιχείου και να αναγνωρίσει όσα εικονοστοιχεία απαιτούνται στην εκάστοτε ανάλυση,
- ✓ Οι ποσοτικοί αλγόριθμοι μπορούν να επεξεργαστούν συγχρόνως πολυδιάστατα δεδομένα, και να χρησιμοποιούν όλα τα επίπεδα της ραδιομετρικής διαχωριστικής ικανότητας.

Από την άλλη πλευρά ως μειονέκτημα της προσέγγισης αυτής χαρακτηρίζεται το γεγονός ότι είναι ανεπαρκής στην απόδοση χωρικών και σημασιολογικών σχέσεων, κάτι που οφείλεται στην δυσκολία με την οποία, αποφάσεις για το σχήμα, μέγεθος, διεύθυνση και υφή μπορούν να εξαχθούν με χρήση των μέχρι σήμερα γνωστών αλγορίθμων. Η πρόοδος με τις νέες τεχνικές της τεχνητής οράσεως και εμπειρικών συστημάτων και τους υπολογιστές παράλληλης επεξεργασίας μπορεί να βοηθήσουν στο να παρακαμφθούν μερικά από αυτά τα προβλήματα.

1.3 Αδυναμίες

Οι διάφοροι αλγόριθμοι κατάτμησης που υφίστανται μπορούν να δώσουν ένα μεγάλο πλήθος διαφορετικών κατατμήσεων, ανάλογα με τις τιμές των παραμέτρων που υπεισέρχονται στην διαδικασία της κατάτμησης. Η αδυναμία που υφίσταται είναι ότι ο χρήστης πρέπει να υλοποιεί δοκιμές με διάφορες τιμές παραμέτρων, κινούμενος στα «τυφλά» (με πολλαπλές δοκιμές και απορρίψεις), όσον αφορά την επιλογή των τιμών των παραμέτρων και επιπλέον δεν μπορεί να ελέγχει το αποτέλεσμα της κατάτμησης με μαθηματικό αναλυτικό τρόπο παρά μόνο οπτικά, σε σύγκριση με την αρχική εικόνα.

1.4 Απαιτούμενη Ανάγκη

Συνεπώς πρέπει να υπάρξει τρόπος, ώστε η επιλογή των τιμών των παραμέτρων να γίνεται με βάση συγκεκριμένη διαδικασία, η οποία να οδηγείται από το εκάστοτε αποτέλεσμα κατάτμησης και μάλιστα από την βελτιστοποίηση του αποτελέσματος αυτού. Ο έλεγχος του αποτελέσματος της κατάτμησης πρέπει να ελέγχεται μαθηματικά με δείκτες ώστε να υπεισέρχεται η έννοια της αντικειμενικότητας και όχι της υποκειμενικότητας του χρήστη, στο αποτέλεσμα της κατάτμησης.

1.5 Σκοπός της Διπλωματικής Εργασίας

Για την εξαγωγή των αντικειμένων από μια εικόνα δηλαδή την κατάτμηση της εικόνας, έχει αναπτυχθεί πλήθος αλγορίθμων σε διάφορες γλώσσες προγραμματισμού καθώς και πακέτα λογισμικών. Σε αυτά απαραίτητο στοιχείο αποτελούν οι τιμές των παραμέτρων που περιλαμβάνονται σε κάθε έναν αλγόριθμο, ώστε να προκύψουν αντικείμενα διαφόρων σχημάτων και μεγεθών.

Οι γενετικοί αλγόριθμοι (ΓΑ) αποτελούν μια μορφή εξελιγμένων και πολύπλοκων αλγορίθμων, οι οποίοι ανήκουν στο πεδίο της τεχνητής νοημοσύνης και χρησιμοποιούνται στην αναζήτηση της βέλτιστης λύσης σε προβλήματα με μεγάλο αριθμό παραμέτρων και χώρο λύσεων με ιδιαίτερως απρόβλεπτη μορφή και συμπεριφορά.

Σκοπός της παρούσας διπλωματικής είναι ο συνδυασμός του πεδίου της τεχνητής νοημοσύνης με την επιστήμη της τηλεπισκόπησης. Συγκεκριμένα εφαρμόζονται ΓΑ σε αλγορίθμους κατάτμησης, με βάση συγκεκριμένους δείκτες ως συνάρτηση αξιολόγησης, με σκοπό να βελτιστοποιηθούν οι παράμετροι που χρησιμοποιεί έκαστος από αυτούς και να προκύψει η βέλτιστη δυνατή κατάτμηση βασισμένη πάνω σε μια εικόνα αναφοράς. Έναυσμα για την υλοποίηση της συγκεκριμένης μεταπτυχιακής διπλωματικής αυτής αποτέλεσε η διπλωματική εργασία του συγγραφέως για την Σχολή Αγρονόμων Τοπογράφων καθώς και η δημοσίευση της επιστημονικής εργασίας των Yong Liu et al (2012) σχετικά με την πρόταση νέων δεικτών κατάλληλων να ελέγχουν την ορθότητα μιας κατάτμησης.

Οι επιμέρους υποστόχοι αφορούν:

- ☞ Τη **διερεύνηση** τριών αλγορίθμων κατάτμησης, βασισμένοι ο πρώτος στην μέθοδο watershed, ο δεύτερος στην μέθοδο region growing και ο τρίτος στην μέθοδο region merging, υλοποιημένους στην γλώσσα προγραμματισμού python, ώστε να κατανοηθεί ο τρόπος λειτουργίας του,
- ☞ Την **εφαρμογή των ΓΑ** σε αυτούς τους αλγορίθμους, με δοκιμές σε διαφορετικές εικόνες, ώστε να καλυφθεί όσο το δυνατόν περισσότερο εύρος πιθανών διαφορετικών προτύπων,
- ☞ Την **εξαγωγή** στατιστικών αποτελεσμάτων σχετικά με την αποτελεσματικότητα της παραπάνω εφαρμογής, πραγματοποιώντας **σύγκριση** ανάμεσα στα εξαγόμενα αποτελέσματα και τα ιδεατά σύμφωνα με την εικόνα αναφοράς, η οποία προκύπτει με διαδικασία η οποία αναφέρεται στο εδάφιο 3.4.

1.6 Δομή της Διπλωματικής Εργασίας

Συνοπτικά, η διπλωματική εργασία περιλαμβάνει τέσσερα κυρίως κεφάλαια συμπεριλαμβανομένου του κεφαλαίου της εισαγωγής (Κεφάλαιο 1), με το εξής περιεχόμενο:

Στο δεύτερο κεφάλαιο, “ΑΝΑΣΚΟΠΗΣΗ ΒΙΒΛΙΟΓΡΑΦΙΑΣ”, παρουσιάζονται και περιγράφονται οι διάφορες έννοιες και αλγόριθμοι που χρησιμοποιούνται στην παρούσα διπλωματική. Συγκεκριμένα:

- Αρχικά γίνεται μια πλήρη περιγραφή του πεδίου των ΓΑ, ώστε να γίνει όσο το δυνατόν περισσότερο κατανοητός τόσο ο τρόπος λειτουργίας τους όσο και να αποσαφηνιστεί η ορολογία που χρησιμοποιείται,

- Ακολουθεί η παρουσίαση της βιβλιοθήκης ΓΑ *ryenolve* που υφίσταται στην γλώσσα προγραμματισμού *rython*. Περιγράφεται πλήρως η εργαλειοθήκη αυτή, με όλες τις δυνατότητες και τις πιθανές επιλογές σε κάθε δυνατότητα που διαθέτει,
- Εν συνεχεία περιγράφονται οι τρεις αλγόριθμοι κατάτμησης που υπεισέρχονται στην λειτουργία του ΓΑ,
- Τέλος αναλύονται κάποιες από τις προσπάθειες εφαρμογής ΓΑ στο πεδίο της τηλεπισκόπησης για την αυτόματη εξαγωγή χαρακτηριστικών, μέχρι σήμερα.

Στο τρίτο κεφάλαιο, “ΜΕΘΟΔΟΛΟΓΙΑ” αναλύεται ο τρόπος προσέγγισης του θέματος και παρουσιάζονται όλα τα αποτελέσματα της εργασίας. Συγκεκριμένα:

- Αρχικά περιγράφεται ο αλγόριθμος ελέγχου και ο κώδικας που αναπτύχθηκε στην γλώσσα *rython* για του υπολογισμό της ακρίβειας των κατατμήσεων με βάση τους δείκτες που χρησιμοποιούνται,
- Ακολουθεί η ανάλυση του ΓΑ καθώς και του κώδικα υλοποίησης του,
- Παρουσιάζονται οι εικόνες που θα χρησιμοποιηθούν στην εργασία,
- Αναλύεται ο τρόπος δημιουργίας των εικόνων αναφοράς, για την υλοποίηση των ΓΑ,
- Τέλος παρουσιάζονται οι δοκιμές εφαρμογής των ΓΑ στους διατιθέμενους αλγορίθμους για κάθε εικόνα και για κάθε δείκτη, καθώς και τα αποτελέσματα που προέκυψαν σε κάθε εφαρμογή.

Στο τέταρτο κεφάλαιο τέλος, “ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΡΟΤΑΣΕΙΣ”, καταγράφονται τα συμπεράσματα και οι προτάσεις, που προέκυψαν από την ενασχόληση με το συγκεκριμένο θέμα.

2. ΑΝΑΣΚΟΠΗΣΗ ΒΙΒΛΙΟΓΡΑΦΙΑΣ

2.1 Γενικά

Στο κεφάλαιο αυτό γίνεται αναφορά στις διάφορες έννοιες που χρησιμοποιούνται στην μεταπτυχιακή εργασία. Συγκεκριμένα αναλύεται η έννοια των ΓΑ, η βιβλιοθήκη των ΓΑ στην γλώσσα προγραμματισμού *python*, οι αλγόριθμοι κατάτμησης που χρησιμοποιούνται και τέλος γίνεται μια αναφορά σε προηγούμενες παρόμοιες μελέτες που έχουν γίνει με την χρησιμοποίηση ΓΑ στην επιστήμη της Τηλεπισκόπησης.

2.2 Γενετικοί Αλγόριθμοι

Πολλά από τα προβλήματα αναζήτησης και βελτιστοποίησης περιλαμβάνουν έναν μεγάλο αριθμό παραμέτρων σχεδίασης, οι οποίες μάλιστα συχνά αλληλεπιδρούν μεταξύ τους, καθώς και έναν χώρο λύσεων με ιδιαίτερως απρόβλεπτη μορφή και συμπεριφορά. Σε τέτοιες περιπτώσεις οι κλασσικοί αλγόριθμοι βελτιστοποίησης κρίνονται ανεπαρκείς αφού είτε αναγκάζονται να ελέγξουν έναν υπερβολικά μεγάλο αριθμό πιθανών λύσεων είτε ενδέχεται να εγκλωβιστούν σε μία λύση η οποία απέχει κατά πολύ από τη βέλτιστη. Είναι απαραίτητη λοιπόν η εφαρμογή πιο εξελιγμένων και πολύπλοκων αλγορίθμων οι οποίοι θα επιτελούν μια πλήρη εξερεύνηση του χώρου λύσεων και θα συγκλίνουν προς μία αρκετά ικανοποιητική λύση. Οι ΓΑ είναι ένα τέτοιο είδος αλγορίθμων οι οποίοι χρησιμοποιούνται όλο και περισσότερο σε πλήθος εφαρμογών.

2.2.1 Γενικά

Η τυπική δομή των ΓΑ έχει περιγραφεί από τον Goldberg (1989). Γενικότερα τα τελευταία τριάντα χρόνια, έχει παρατηρηθεί ένα συνεχώς αυξανόμενο ενδιαφέρον για την ανάπτυξη συστημάτων επίλυσης προβλημάτων βασισμένων στις αρχές της Γενετικής Εξέλιξης και της Κληρονομικότητας. Η κεντρική ιδέα πίσω από τους γενετικούς αλγορίθμους είναι η μίμηση της βιολογικής διεργασίας της εξέλιξης. Σύμφωνα με τον κανόνα της φυσικής επιλογής, οι οργανισμοί που δε μπορούν να επιβιώσουν στο περιβάλλον τους πεθαίνουν, ενώ οι υπόλοιποι πολλαπλασιάζονται μέσω της αναπαραγωγής.

Τα μειονεκτήματα των κλασσικών μεθόδων αναζήτησης και βελτιστοποίησης όπως μικρή αποδοτικότητα σε προβλήματα με πλήθος παραμέτρων, ύπαρξη περιορισμών σε πολλά προβλήματα, χρησιμοποίηση ενός μόνο σημείου του χώρου αναζήτησης κάθε φορά, χρησιμοποίηση πλήθους πληροφορίας και όχι ποιοτικής πληροφορίας, κ.α. καθώς και η συνεχώς αυξανόμενη ανάγκη για παραγωγή λογισμικού που να μπορεί να εκμεταλλεύεται πιο αποδοτικά τις τεράστιες δυνατότητες του υλικού υπολογιστών, ήταν η βασική αιτία που ώθησε τους επιστήμονες σ' αυτήν την αναζήτηση. Αυτού του είδους τα συστήματα λειτουργούν διατηρώντας ένα πληθυσμό κωδικοποιημένων πιθανών λύσεων και εφαρμόζοντας πάνω σε αυτές διάφορους γενετικούς τελεστές. Οι τελεστές αυτοί αντιγράφουν τον τρόπο με τον οποίο αναπαράγονται και μεταλλάσσονται τα χρωμοσώματα των κυττάρων των ζωντανών οργανισμών. Έτσι, περνώντας από γενιά σε γενιά, τα συστήματα αυτά δημιουργούν συνεχώς νέους πληθυσμούς πιθανών λύσεων χρησιμοποιώντας, στοιχεία τόσο από την προηγούμενη γενιά, όσο και εντελώς καινούρια τα οποία δοκιμάζονται για τυχόν καλή απόδοσή τους.

Επανειλημμένες δοκιμές και πειράματα έχουν δείξει ότι μια “φυσική” αναπαράσταση των πιθανών λύσεων για ένα δεδομένο πρόβλημα, σε συνδυασμό με την εφαρμογή σε αυτή μιας οικογένειας γενετικών τελεστών, αποτελεί πολύ χρήσιμο εργαλείο στην προσπάθεια προσέγγισης των πραγματικών λύσεων σε μια πολύ μεγάλη ποικιλία προβλημάτων και

εφαρμογών. Αυτό το γεγονός, μετατρέπει αυτή τη “φυσικού μοντέλου” προσέγγιση σε μια πολλά υποσχόμενη κατεύθυνση, όσον αφορά την επίλυση προβλημάτων γενικότερα.

2.2.2 Ιστορική Αναδρομή

Η πρώτη εμφάνιση των ΓΑ χρονολογείται στις αρχές του 1950, όταν διάφοροι βιολόγοι επιστήμονες αποφάσισαν να χρησιμοποιήσουν υπολογιστές στην προσπάθειά τους να προσομοιάσουν πολύπλοκα βιολογικά συστήματα. Η συστηματική τους ανάπτυξη όμως, που οδήγησε στην μορφή με την οποία είναι γνωστοί και σήμερα, πραγματοποιήθηκε στις αρχές του 1970 από τον Holland και τους συνεργάτες του στο Πανεπιστήμιο του Michigan. Ο αρχικός στόχος του Holland ήταν να κάνει μια αυστηρή εξήγηση των διαδικασιών προσαρμογής των φυσικών συστημάτων καθώς και να σχεδιάσει τεχνητά συστήματα λογισμικού που διατηρούν τους σημαντικούς μηχανισμούς των φυσικών συστημάτων.

Ο Bagley (1967) με την διδακτορική του διατριβή ουσιαστικά «βαφτίζει» τους γενετικούς αλγόριθμους.

Ο Rosenberg (1967) δημοσιεύει εργασία, στην οποία γίνεται λόγος για προσομοίωση πληθυσμών μονοκύτταρων οργανισμών σε υπολογιστικό περιβάλλον.

Ο Holland (1975) εκδίδει το βιβλίο "Προσαρμογή στα Φυσικά και Τεχνητά συστήματα", στο οποίο αναπτύσσει τις ιδέες και την θεωρία των ΓΑ. Το βιβλίο θεωρείται πλέον κλασικό για τον χώρο. Θίγονται θέματα όπως η θεωρία των σχημάτων, η βέλτιστη κατανομή των ευκαιριών, σχέδια αναπαραγωγής, γενετικές λειτουργίες, η ευρωστία των ΓΑ και πλήθος άλλα.

Ο De Jong (1975) με την εργασία που εκδίδει βοηθά την πειραματική αξιολόγηση των ΓΑ. Σύμφωνα με αυτήν, προτείνονται λειτουργίες που ελέγχουν έναν ΓΑ και την ικανότητά του να αντιμετωπίζει δύσκολα προβλήματα.

Ο Grefenstette (1980) δημιουργεί το GENESIS, ένα σύστημα ανάπτυξης ΓΑ υλοποιημένο στη γλώσσα προγραμματισμού C, που έχει βοηθήσει σημαντικά στη διάδοση του γενετικού προγραμματισμού καθώς έγινε διαθέσιμο στο ευρύ κοινό.

Το 1985 πραγματοποιείται το 1ο Διεθνές Συνέδριο των ΓΑ και των εφαρμογών τους. Μετά από αυτό ο χώρος των ΓΑ αποκτά ένα μεγάλο συνέδριο που πλέον λαμβάνει χώρα κάθε δύο χρόνια και αντικατοπτρίζει το μεγάλο οργασμό που παρατηρείται σε επίπεδο τόσο θεωρίας, όσο και εφαρμογών.

Πολυάριθμες εκδόσεις βιβλίων για ΓΑ, 1989-1999 όπως «Genetic Algorithms in Search, Optimization», and Machine Learning», «An introduction to Genetic Algorithms for Scientists and Engineers», «Adaptive Learning by Genetic Algorithms», «An Introduction to Genetic Algorithms» κ.α. . Άλλη μια ένδειξη της τεράστιας ανάπτυξης του χώρου και της αποδοχής της νέας τεχνολογίας.

Ανάπτυξη πακέτων λογισμικού για ΓΑ, 1990-1999. Πολλές εταιρίες δημιουργούν εμπορικά πακέτα που επιτρέπουν σε χρήστες να ενσωματώσουν στις εφαρμογές τους στοιχεία Γενετικού Προγραμματισμού (Genetic Programming). Ένα τέτοιο πακέτο είναι το EOS (Evolutionary Object System). Βασίζεται στη δημοφιλή γλώσσα αντικειμενοστραφούς προγραμματισμού C++ και παρέχει μεγάλες δυνατότητες προσαρμογών και επεκτάσεων.

Από το 2000 έως και σήμερα έχουν δημιουργηθεί πλήθος εφαρμογών βασισμένες στους ΓΑ. Πολλές επιστημονικές δημοσιεύσεις τείνουν να επιλύσουν καθημερινά προβλήματα με την βοήθεια των ΓΑ. Η ενσωμάτωση τους στην επίλυση καθημερινών προβλημάτων αποτελεί πλέον γεγονός.

2.2.3 Ορολογία Γενετικών Αλγορίθμων

Όπως αναφέρθηκε παραπάνω οι ΓΑ έχουν προκύψει μέσα από τον χώρο της βιολογίας και της γενετικής οπότε και δανείζονται τις ονομασίες των διαφόρων στοιχείων τους από αυτό τον χώρο. Συγκεκριμένα λοιπόν (Λυκοθανάσης, 2001):

Γονίδια (Genes) και Χρωμοσώματα ή Άτομα (Individuals): Γονίδιο είναι μια πιθανή τιμή που μπορεί να λάβει μια μεταβλητή. Ένα χρωμόσωμα αποτελείται από γονίδια, δηλαδή από ένα σύνολο πιθανών τιμών που μπορεί να λάβουν οι μεταβλητές ενός προβλήματος. Ένα συγκεκριμένο χρωμόσωμα το οποίο είναι και μοναδικό μπορεί να αποκωδικοποιηθεί σε ένα σύνολο παραμέτρων που αναπαριστούν μία πιθανή λύση του προβλήματος. Το χρωμόσωμα μπορεί να είναι μια σειρά πραγματικών ή δυαδικών αριθμών ή ακόμα και συνδυασμοί αυτών των δυο.

Αρχικός Πληθυσμός (Initial Population): Είναι ο αριθμός των αρχικών χρωμοσωμάτων από τα οποία θα ξεκινήσει η διαδικασία αναζήτησης του ΓΑ. Όσο μεγαλύτερο είναι το πλήθος του αρχικού πληθυσμού τόσο πολυπλοκότερη γίνεται η επίλυση του προβλήματος αλλά με περισσότερες πιθανότητες εύρεσης της βέλτιστης λύσης και με λιγότερες πιθανότητες ο αλγόριθμος να θεωρήσει ένα τοπικό ελάχιστο ως το ολικό ελάχιστο δηλαδή να οδηγηθεί σε λανθασμένη λύση.

Πληθυσμοί (Population) και Γενεές (Generations) ΓΑ: Σε κάθε βήμα του αλγορίθμου, και με βάση τον αρχικό πληθυσμό δημιουργείται ένας νέος πληθυσμός, ο οποίος αποτελεί και την επόμενη γενιά του αλγορίθμου. Στην εκάστοτε νέα γενιά γίνεται αναζήτηση για εντοπισμό του χρωμοσώματος που μπορεί να δώσει το βέλτιστο αποτέλεσμα. Η διαδικασία τερματίζεται είτε με κατώφλι ένα αριθμό γενεών είτε στο καλύτερο χρωμόσωμα είτε με διάφορα άλλα κριτήρια τα οποία θα αναφερθούν παρακάτω.

Γονείς (Parents) και Τέκνα (Children): Τα ζεύγη χρωμοσωμάτων που επιλέγονται σε κάθε γενιά ώστε να διασταυρωθούν μεταξύ τους και να παραγάγουν απογόνους ονομάζονται γονείς ενώ οι απόγονοι τέκνα. Τα τέκνα μπορούν να δημιουργηθούν είτε μέσω της διαδικασίας της διασταύρωσης είτε μέσω της μετάλλαξης.

Διασταύρωση (Crossover): Συνδυασμός δυο γονέων ώστε να προκύψει ένα νέο τέκνο. Η διασταύρωση παρουσιάζεται με μια πιθανότητα p_c να συμβεί και είναι δυνατό να πραγματοποιηθεί σε ένα ή περισσότερα σημεία των χρωμοσωμάτων - γονέων.

Μετάλλαξη (Mutation): Οι μεταλλάξεις είναι χαμηλής πιθανότητας p_m τυχαίες αλλαγές στα άτομα κάθε γενιάς ώστε να δημιουργηθούν τέκνα χωρίς διασταύρωση. Η μετάλλαξη παρέχει τη γενετική ποικιλομορφία και επιτρέπει στο γενετικό αλγόριθμο να ψάξει βέλτιστη λύση σε ένα ευρύτερο διάστημα.

Καταλληλότητα (Fitness): Το βασικότερο στοιχείο για την λειτουργία ενός ΓΑ. Η αποτελεσματικότητα του ΓΑ εξαρτάται από την σωστή και εύστοχη επιλογή της συνάρτησης καταλληλότητας ή αξιολόγησης. Συνήθως το πεδίο τιμών της παραπάνω συνάρτησης είναι από 0 έως 1, αναλόγως βέβαια και του προβλήματος, με την τιμή 1 να

θεωρείται ότι έχει ευρεθεί το τέλειο χρωμόσωμα άρα και η βέλτιστη λύση ενώ η τιμή 0 ότι η λύση δεν έχει καμιά ευστοχία. Γενικότερα πάντως στα πακέτα που χρησιμοποιούν ΓΑ ελαχιστοποίηση της συνάρτησης αξιολόγησης συνεπάγεται βέλτιστη λύση του προβλήματος (π.χ. στην εργαλειοθήκη ΓΑ του matlab, ενώ στην βιβλιοθήκη ΓΑ της python, την pyevolne, ορίζεται αν η βελτιστοποίηση αναφέρεται σε ελαχιστοποίηση ή μεγιστοποίηση της συνάρτησης αξιολόγησης).

2.2.4 Κύρια Χαρακτηριστικά ενός Γενετικού Αλγορίθμου

Οι ΓΑ όπως αναφέρθηκε ήδη, πλεονεκτούν στην λύση προβλημάτων αναζήτησης και βελτιστοποίησης σε σχέση με τις παραδοσιακές μεθόδους. Τα κυριότερα χαρακτηριστικά που επιδρούν στην υπεροχή τους αυτή, σύμφωνα με τον Goldberg (1989), είναι τα εξής:

1. *Οι ΓΑ κωδικοποιούν το σύνολο των τιμών που μπορούν να πάρουν οι μεταβλητές του προβλήματος και χρησιμοποιούν αυτή την κωδικοποίηση και όχι τις μεταβλητές καθαυτές για την επίλυση:* Συγκεκριμένα απαιτούν το σύνολο των φυσικών παραμέτρων της βελτιστοποίησης, να κωδικοποιηθεί σε συμβολοσειρές πεπερασμένου μήκους, κάνοντας χρήση ενός πεπερασμένου αλφαβήτου. Για παράδειγμα, αναφέρεται το εξής πρόβλημα βελτιστοποίησης: Έστω ένα μαύρο κουτί με πέντε δυαδικούς διακόπτες (on-off). Για κάθε συνδυασμό των διακοπών s παράγεται μία έξοδος $f(s)$. Ζητείται ο συνδυασμός των διακοπών που μεγιστοποιεί την έξοδο. Με τις παραδοσιακές μεθόδους, το μέγιστο θα εντοπιζόταν με "παίξιμο" των διακοπών πηγαίνοντας από συνδυασμό σε συνδυασμό με ψάξιμο στα τυφλά, καθ' ότι δεν είναι γνωστός ο τύπος της συνάρτησης. Σε ένα ΓΑ όμως, η πρώτη ενέργεια είναι η κωδικοποίηση των διακοπών ως συμβολοσειρές πεπερασμένου μήκους. Μια απλή κωδικοποίηση θα μπορούσε να γίνει θεωρώντας μια δυαδική συμβολοσειρά μήκους πέντε, όπου η κάθε θέση αναπαριστά ένα διακόπτη. Το 0 αντιστοιχεί στη θέση off και το 1 στη θέση on. Δηλαδή, η συμβολοσειρά 11110 κωδικοποιεί το συνδυασμό κατά τον οποίο οι πρώτοι τέσσερις διακόπτες είναι on και ο τελευταίος off. Η κωδικοποίηση δεν είναι απαραίτητο να είναι πάντα δυαδική. Όπως θα φανεί και αργότερα, μπορεί να γίνει με πολλούς τρόπους, αρκετοί από τους οποίους ίσως και να μην είναι προφανείς. Το στοιχείο της κωδικοποίησης, όπως εξηγείται παρακάτω, είναι εκείνο που επιτρέπει στους ΓΑ να κάνουν παράλληλη επεξεργασία δεδομένων.
2. *Οι ΓΑ κάνουν αναζήτηση σε πολλά σημεία ταυτόχρονα και όχι μόνο σε ένα:* Στις περισσότερες μεθόδους βελτιστοποίησης, η επεξεργασία γίνεται βήμα προς βήμα, πηγαίνοντας προσεκτικά από σημείο σε σημείο του πεδίου ορισμού του προβλήματος. Ωστόσο με τον τρόπο αυτό δημιουργούνται κίνδυνοι, ο κυριότερος από τους οποίους είναι να περιοριστεί η αναζήτηση σε μια περιοχή τοπικού ακρότατου, που δεν είναι ολικό. Οι ΓΑ εξαλείφουν αυτόν τον κίνδυνο ενεργώντας ταυτόχρονα πάνω σε ένα ευρύ σύνολο σημείων (σύνολο από συμβολοσειρές). Έτσι μπορούν να "ανεβαίνουν" πολλούς λόφους (hill climbing) την ίδια στιγμή, ελαχιστοποιώντας την πιθανότητα να βρουν μια λάθος κορυφή. Επιστρέφοντας στο παράδειγμα με το μαύρο κουτί, οι κλασσικές μέθοδοι θα ξεκινούσαν το ψάξιμο από ένα συνδυασμό των διακοπών και στη συνέχεια, εφαρμόζοντας κάποιο κανόνα μετάβασης, θα δοκίμαζαν τον επόμενο (ψάξιμο δηλαδή σημείο προς σημείο). Αντιθέτως, ένας ΓΑ αρχίζει το ψάξιμο του από ένα πληθυσμό συνδυασμών συμβολοσειρών και κατόπιν παράγει διαδοχικά καινούριους. Ένας αρχικός πληθυσμός θα μπορούσε να είναι, π.χ. 01101, 11000, 01000 και 10011. Έπειτα, "τρέχοντας" ο αλγόριθμος δημιουργεί νέους πληθυσμούς που σιγά σιγά συγκλίνουν προς την επιθυμητή λύση. Διαλέγοντας ένα

πληθυσμό που να καλύπτει αντιπροσωπευτικά ένα μεγάλο εύρος τιμών μπορούν να προκύψουν ικανοποιητικά αποτελέσματα.

3. *Οι ΓΑ χρησιμοποιούν μόνο την αντικειμενική συνάρτηση και καμία επιπρόσθετη πληροφορία:* Η αναζήτηση για την εύρεση των βέλτιστων τιμών των παραμέτρων είναι κατά κάποιο τρόπο “τυφλή” καθώς αξιοποιεί μόνο όση πληροφορία περιέχεται στην αντικειμενική συνάρτηση πράγμα που της προσδίδει μεγάλη ευελιξία. Αντίθετα οι περισσότερες κοινές μέθοδοι αναζήτησης απαιτούν αρκετές βοηθητικές πληροφορίες για τη συνάρτηση που επεξεργάζονται. Τέτοιου είδους πληροφορίες δεν προαπαιτούνται από τους ΓΑ. Έτσι όμως, προκύπτει το ερώτημα αν συμφέρει να αγνοούνται οι βοηθητικές πληροφορίες. Γι' αυτό ακριβώς το λόγο έχουν αναπτυχθεί μορφές ΓΑ που αξιοποιούν και τέτοιου είδους πληροφορίες, δηλαδή χρησιμοποιούν υπάρχουσα γνώση που υπάρχει για την συγκεκριμένη συνάρτηση, την οποία και συνδυάζουν με την εφαρμογή ΓΑ. (Knowledge - Based Genetic Algorithms).
4. *Οι ΓΑ χρησιμοποιούν πιθανοθεωρητικούς κανόνες μετάβασης και όχι ντετερμινιστικούς:* Η χρήση πιθανοθεωρητικών κανόνων μετάβασης είναι κυρίαρχο γνώρισμα των ΓΑ, χωρίς αυτό να σημαίνει ότι η όλη διαδικασία βαδίζει στην τύχη. Δε λαμβάνονται αποφάσεις στην τύχη. Το στοιχείο της τύχης χρησιμοποιείται ως οδηγός για αναζήτηση σε περιοχές που αναμένεται να δώσουν καλά αποτελέσματα.

2.2.5 Βασικά Στοιχεία ενός Γενετικού Αλγορίθμου

Ένας αλγόριθμος πρέπει να έχει κάποια βασικά στοιχεία ώστε να θεωρείται γενετικός. Τα απαραίτητα συστατικά τα οποία τον συνδέουν με το προς επίλυση πρόβλημα, είναι η κωδικοποίηση των πιθανών λύσεων και η αντικειμενική συνάρτηση (Λυκοθανάσης, 2001).

Κωδικοποίηση (Coding): Κύριος στόχος της κωδικοποίησης είναι να αναπαριστά με ικανοποιητικό τρόπο τα επιμέρους χαρακτηριστικά των λύσεων, ώστε να διευκολύνει τις επόμενες λειτουργίες του αλγορίθμου (κυρίως την επιλογή). Η κωδικοποίηση που αφορά ένα σύνολο πιθανών λύσεων του προβλήματος πρέπει να γίνει με ένα μαθηματικό, φορμαλιστικό τρόπο, ώστε να είναι δυνατή η επεξεργασία από τον υπολογιστή. Εξάλλου, κωδικοποίηση υπάρχει και στο φυσικό μοντέλο (χρωμοσώματα) και μάλιστα όλες οι αλλαγές που παρατηρούνται στους οργανισμούς γίνονται πάνω στα κωδικοποιημένα χαρακτηριστικά των χρωμοσωμάτων.

Η κωδικοποίηση μπορεί να γίνει με διάφορους τρόπους αναλόγως και του προβλήματος. Η πιο απλή κωδικοποίηση είναι με δυαδικά ψηφία (bits). Στην περίπτωση αυτή κάθε λύση αναπαρίσταται από μια δυαδική συμβολοσειρά (binary string) καθορισμένου μήκους. Πάντως, έχουν αναφερθεί ποικίλες μορφές κωδικοποιήσεων, που καθεμία εξαρτάται από το υπό εξέταση πρόβλημα. Δεν υπάρχει κάποια κωδικοποίηση η οποία να θεωρείται αποτελεσματική για κάθε είδους πρόβλημα, όπως επίσης είναι δυνατόν κάποιο πρόβλημα να επιδέχεται περισσότερες από μια κωδικοποιήσεις. Το σίγουρο είναι ότι η κωδικοποίηση είναι το κρίσιμο αρχικό βήμα στην εφαρμογή του ΓΑ και, αν δεν είναι προσεκτική, πιθανότατα θα αποβεί μοιραία για την επιτυχία του. Η καταλληλότητα της κωδικοποίησης εξαρτάται σε μεγάλο βαθμό από τη διαίσθηση που αισθάνεται για το συγκεκριμένο πρόβλημα ο σχεδιαστής αλλά και από την πείρα του η οποία έχει προκύψει από την ενασχόληση με παρόμοια προβλήματα. Συμβαίνει μερικές φορές, μάλιστα, προφανείς τρόποι κωδικοποίησης να μην είναι αρκετά αποτελεσματικοί. Σκεπτόμενος λοιπόν κάποιος όλα τα παραπάνω μπορεί να προβεί στην κρίσιμη ερώτηση για το ποιοι είναι οι

παράγοντες εκείνοι οι οποίοι καθορίζουν το είδος της κωδικοποίησης που πρέπει να επιλεγεί για ένα συγκεκριμένο πρόβλημα. Όμως ξεκάθαρη απάντηση δεν μπορεί να δοθεί η οποία να καλύπτει κάθε δυνατή περίπτωση. Μερικές γενικού τύπου συμβουλές θα φανούν στην παραπέρα ανάπτυξη του θέματος, με τη βοήθεια παραδειγμάτων.

Ένα παράδειγμα το οποίο μπορεί να ξεκαθαρίσει περαιτέρω της έννοια της κωδικοποίησης είναι το εξής:

Έστω η συνάρτηση $f(x) = x^3$, $x \in [0, 63]$ και x : ακέραιος. Ζητείται το μέγιστο της συνάρτησης στο πεδίο ορισμού της. Για να λυθεί το πρόβλημα από ένα ΓΑ πρέπει να επινοηθεί ένας τρόπος κωδικοποίησης των πιθανών λύσεων. Ο πιο προφανής και τελικά, όπως θα αποδειχθεί, πιο αποτελεσματικός τρόπος κωδικοποίησης είναι να αναπαρασταθεί η κάθε λύση με μια δυαδική συμβολοσειρά μήκους 6, που αριθμητικά θα ισοδυναμεί με την αντίστοιχη δεκαδική τιμή της λύσης. Έτσι καλύπτεται όλο το πεδίο ορισμού $[0,63]$ από τις 64 δυνατές συμβολοσειρές (000000 - 111111) αυτού του είδους. Π.χ. η συμβολοσειρά 001001 αντιστοιχεί, κατά τα γνωστά, στην τιμή 9 του δεκαδικού συστήματος. Συνήθως, σε προβλήματα βελτιστοποίησης μαθηματικών συναρτήσεων, η δυαδική είναι η πιο βολική και αποδοτική κωδικοποίηση.

Αντικειμενική Συνάρτηση ή Συνάρτηση Αξιολόγησης (Fitness Function): Το επόμενο ουσιαστικό στοιχείο για τον ΓΑ σε σχέση με το πρόβλημα που καλείται να αντιμετωπίσει, είναι η αντικειμενική συνάρτηση. Αποτέλεσμα της συνάρτησης αυτής είναι μια τιμή, ανάλογη του πόσο καλά λύνεται το πρόβλημα, για κάθε αποκωδικοποιημένη συμβολοσειρά που λαμβάνει. Η τιμή αυτή αποτελεί και τον καθοριστικό παράγοντα επιβίωσης και πολλαπλασιασμού ή όχι του ατόμου. Η αντικειμενική συνάρτηση παίζει το ρόλο του περιβάλλοντος στο τεχνητό μοντέλο. Ουσιαστικά, είναι η μόνη πληροφορία που δέχεται ο αλγόριθμος για το πρόβλημα που λύνει. Όσο πιο εύκολα υπολογίσιμη είναι η συνάρτηση αυτή τόσο πιο γρήγορα εκτελείται ο αλγόριθμος.

Για παράδειγμα, αν εξετάσουμε και πάλι την συνάρτηση $f(x) = x^3$, $x \in [0, 63]$, είναι κατανοητό ότι η αντικειμενική συνάρτηση είναι η ίδια η f , η μεγιστοποίηση της οποίας αποτελεί το ζητούμενο. Έτσι, σε κάθε λύση, δηλαδή σε κάθε πιθανή τιμή της μεταβλητής x , αντιστοιχεί μια *τιμή ικανότητας* ή *απόδοσης (fitness ή score)*, μια τιμή που αξιολογεί το πόσο καλή είναι η λύση για τη μεγιστοποίηση της συνάρτησης και που, για αυτή την περίπτωση είναι η αυτή συνάρτηση.

Με τον καθορισμό της κωδικοποίησης και της αντικειμενικής συνάρτησης, πλέον, ορίζεται το πρόβλημα και ολοκληρώνεται το πρώτο στάδιο εφαρμογής ενός ΓΑ. Αξίζει να σημειωθεί η *αυτονομία* και *ανεξαρτησία* αυτού του σταδίου από τα επόμενα μέρη. Οι λειτουργίες που ακολουθούν από εδώ και πέρα δεν εξαρτώνται από το πώς γίνεται η αναπαραστάση των ατόμων στο τεχνητό περιβάλλον και με ποιο τρόπο αξιολογούνται οι ικανότητές τους. Αυτό είναι σπουδαίο χαρακτηριστικό, διότι επιτρέπει την διαπραγμάτευση πολλών προβλημάτων με μια απλή αλλαγή στην αντικειμενική συνάρτηση, ίσως και στην κωδικοποίηση. Η φάση ορισμού της κωδικοποίησης και της αντικειμενικής συνάρτησης υπάρχουν πάντα σε κάθε ΓΑ, ανεξαρτήτως του προβλήματος.

Ακολουθεί η ανάλυση των γενετικών διαδικασιών που λαμβάνουν χώρα κατά την υλοποίηση ενός ΓΑ, μετά το πέρας των δυο προηγούμενων βημάτων. Σε αυτές τις διαδικασίες εκτελείται ο κύριος όγκος της εργασίας και παράγεται το αποτέλεσμα της βελτιστοποίησης.

Αρχικοποίηση (Initialization): Στο βήμα αυτό επιλέγεται το πλήθος και οι τιμές του αρχικού πληθυσμού, πάνω στον οποίο θα λάβουν χώρα οι λειτουργίες του ΓΑ. Το μέγεθος του πληθυσμού αυτού ορίζεται από τον χρήστη, είναι φυσικά μικρότερο από το δυνατό πλήθος τιμών της συνάρτησης του προβλήματος, ενώ η τιμή του κάθε χρωμοσώματος επιλέγεται με τυχαίο τρόπο ανάμεσα σε όλες τις δυνατές τιμές των μεταβλητών του προβλήματος. Σε μερικές υλοποιήσεις, η επιλογή των αρχικών σημείων γίνεται με ευρετικές μεθόδους, δίνοντας εξαρχής ένα πλεονέκτημα στην αναζήτηση. Έστω ότι επιλέγεται στο παραπάνω πρόβλημα ($f(x) = x^3$, $x \in [0, 63]$), αρχικός πληθυσμός 8 ατόμων. Το κάθε άτομο θα έχει μήκος 6 δυαδικών ψηφίων, άρα πρέπει να επιλεχτούν τυχαία 48 δυαδικά ψηφία τα οποία μπορεί να προκύψουν από 48 διαδοχικές ρίψεις ενός τίμιου νομίσματος. Ένα πιθανό σενάριο θα μπορούσε να βγάλει τις συμβολοσειρές 011011, 111000, 011000, 000000, 100000, 100001, 101010 και 110011. Αφού προκύψει η πρώτη γενιά, ο ΓΑ εισέρχεται στο επαναληπτικό μέρος του. Ο πληθυσμός αυτός πρέπει στην συνέχεια να αξιολογηθεί, δηλαδή να μετρηθεί η ικανότητα επιβίωσης του κάθε ατόμου χωριστά.

Αποκωδικοποίηση (Decoding): Προκειμένου να μπορέσει ο εκάστοτε πληθυσμός είτε της πρώτης είτε οποιασδήποτε γενιάς να αξιολογηθεί πρέπει πρώτα να γίνει η αποκωδικοποίηση του. Ο παραλληλισμός με το φυσικό μοντέλο, ίσως βοηθά στην κατανόηση αυτής της διαδικασίας: Στη φύση τα χρωμοσώματα ενός οργανισμού έχουν στα γονίδιά τους κωδικοποιημένα τα χαρακτηριστικά τους. Το σύνολο αυτής της κωδικοποιημένης γενετικής πληροφορίας ονομάζεται, γονότυπος. Ο γονότυπος δεν είναι αντιληπτός με τις φυσικές αισθήσεις των έμβιων όντων. Αντίθετα, αντιληπτή γίνεται η αλληλεπίδραση του με το περιβάλλον, που έχει ως αποτέλεσμα την ορατή εμφάνιση των χαρακτηριστικών αυτών. Ανάλογος είναι ο ρόλος της αποκωδικοποίησης στο τεχνητό μοντέλο. Εδώ το ρόλο του γονότυπου παίζει η δομή της συμβολοσειράς με τα δυαδικά ψηφία ως αντίστοιχα των γονιδίων π.χ. 110001. Ο φαινότυπος αναφέρεται στην παρατηρήσιμη εμφάνιση μιας συμβολοσειράς, στο πώς φαίνεται στο περιβάλλον της. Περιβάλλον, όμως, θεωρείται η αντικειμενική συνάρτηση, άρα ο φαινότυπος μιας συμβολοσειράς αντιστοιχεί στην αποκωδικοποιημένη τιμή της δηλαδή $49 = 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^0$, που ανήκει στο σύνολο ορισμού της αντικειμενικής συνάρτησης.

Υπολογισμός Ικανότητας ή Αξιολόγησης (Fitness Value): Εφόσον λοιπόν πραγματοποιηθεί η αποκωδικοποίηση, τότε υπολογίζεται για κάθε αποκωδικοποιημένο άτομο η ικανότητα του για επιβίωση. Στη φύση οι ικανότητες των ατόμων δεν είναι προσδιορίσιμες με αυστηρό τρόπο. Είναι, όμως, καθορισμένες από το γενετικό υλικό των χρωμοσωμάτων τους. Εύκολα, πάντως, θα μπορούσε κανείς να ισχυριστεί, π.χ. για τα ζώα ότι μεγαλύτερη τύχη για επιβίωση έχουν όσα μπορούν να ξεφύγουν από αρπαγές, να αντέχουν σε αρρώστιες και γενικά να αντιπαρέρχονται τις όποιες αντιξοότητες παρουσιάζονται κατά τη διάρκεια της ζωής τους. Συνεπώς, ο υπολογισμός της ικανότητας είναι θεμελιώδης λειτουργία για το ΓΑ. Η εφαρμογή της είναι πολύ απλή ιδίως για απλά προβλήματα όπου για κάθε αποκωδικοποιημένη συμβολοσειρά του τρέχοντος πληθυσμού υπολογίζεται η απόδοσή της από την ήδη γνωστή αντικειμενική συνάρτηση. Σε πιο σύνθετα προβλήματα, ο υπολογισμός ικανότητας μπορεί να ισοδυναμεί με την εκτέλεση μιας εργαστηριακής προσομοίωσης.

Αναπαραγωγή (Creating Next Generation): Η διαδικασία αυτή αποτελεί την σημαντικότερη λειτουργία του ΓΑ. Η δομή της είναι σύνθετη και αποτελείται από την διασταύρωση και μετάλλαξη. Για να πραγματοποιηθούν όμως αυτές πρέπει

προηγουμένως να έχει γίνει η επιλογή των ατόμων που θα αποτελέσουν τους γονείς, ώστε να δημιουργηθεί, με βάση αυτά, η επόμενη γενιά.

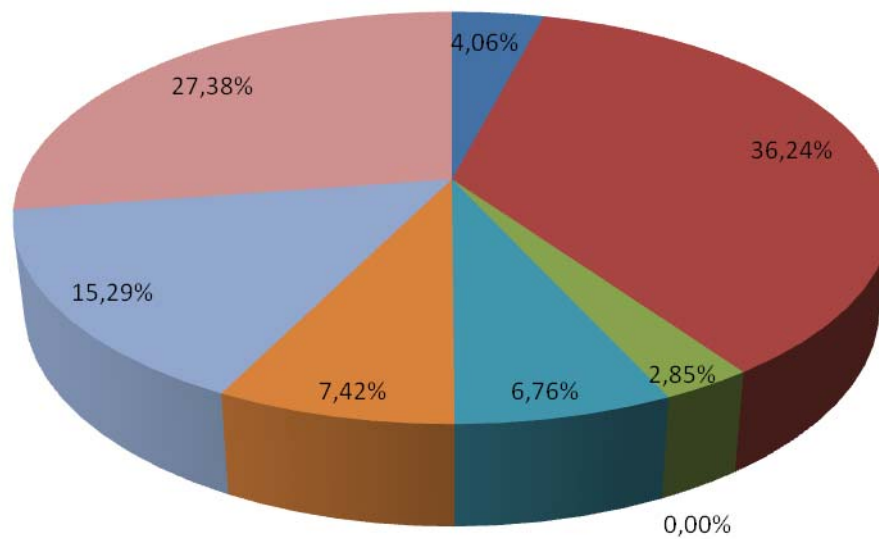
- **Επιλογή (Selection):** Με την επιλογή, βρίσκει εφαρμογή στα πλαίσια του αλγορίθμου, ο νόμος της επιβίωσης του ικανότερου. Ουσιαστικά επιλέγονται τα άτομα τα οποία θα λάβουν μέρος στην αναπαραγωγή και θα κληροδοτήσουν στην επόμενη γενιά μέρος ή το σύνολο των χαρακτηριστικών τους. Στόχος της λειτουργίας της επιλογής είναι να επιτρέπει εκθετική αύξηση των ικανότερων ατόμων και τελικά, μετά από αναπαραγωγή αρκετών γενεών, την επικράτησή τους. ΓΑ χωρίς επιλογή στην αναπαραγωγική του διαδικασία ισοδυναμεί με τυχαίο ψάξιμο. Υπάρχουν διάφορες τεχνικές επιλογής κάθε μια από τις οποίες για να μπορεί να κριθεί θετικά, οφείλει να δίνει με κάποιο τρόπο, μεγαλύτερες πιθανότητες αναπαραγωγής σε άτομα που αξιολογούνται μέσα στο τεχνητό περιβάλλον ως τα πιο ικανά. Ο τελεστής αναπαραγωγής μπορεί να εκφραστεί σε αλγοριθμική βάση, με πολλούς τρόπους. Ο ευκολότερος από αυτούς θα μπορούσε να χαρακτηριστεί η έκφραση μέσω μιας εξαναγκασμένης ρουλέτας, στην οποία κάθε συμβολοσειρά ενός πληθυσμού αντιπροσωπεύεται σε ένα μέρος της ρουλέτας, σε αναλογία με την απόδοσή της.

Προκειμένου να εξηγηθεί περισσότερο η χρήση της εξαναγκασμένης ρουλέτας θεωρείται ο πληθυσμός των 8 ατόμων που δημιουργήθηκε στο προγενέστερο παράδειγμα 011011, 111000, 011000, 000000, 100000, 100001, 101010 και 110011. Υπολογίζεται η απόδοση κάθε ατόμου με βάση την συνάρτηση αξιολόγησης, και σε ποσοστό οπότε προκύπτουν τα εξής:

A/A	Συμβολοσειρά	Αποκωδικο-ποίηση	Απόδοση ($f(x) = x^3$)	Απόδοση %	Συσσωρευμένη πιθανότητα
1	011011	27	19683	4,06	0,0406
2	111000	56	175616	36,24	0,4030
3	011000	24	13824	2,85	0,4316
4	000000	0	0	0	0,4316
5	100000	32	32768	6,76	0,4992
6	100001	33	35937	7,42	0,5734
7	101010	42	74088	15,29	0,7262
8	110011	51	132651	27,38	1,0000
Σύνολο			484567	100	

Πίνακας 2.1- Υπολογισμός απόδοσης για τον αρχικό πληθυσμό

Η παραπάνω αντιστοιχία στην εξαναγκασμένη ρουλέτα έχει ως εξής:



Εικόνα 2.1- Σχηματική αναπαράσταση εξαναγκασμένης ρουλέτας

Τώρα είναι η ώρα να περιστραφεί η ρουλέτα 8 φορές έτσι ώστε σε κάθε περιστροφή να επιλέγεται και ένα άτομο για το νέο πληθυσμό, ο οποίος θα χρησιμοποιηθεί στην διαδικασία της αναπαραγωγής. Έστω ότι έχει παραχθεί η εξής ακολουθία 8 τυχαίων αριθμών στο διάστημα $[0,1]$:

0.5663, 0.2358, 0.3569, 0.8597, 0.1234, 0.4212, 0.7187, 0.4567.

Για κάθε ένα από αυτούς τους αριθμούς εξετάζεται σε ποιο διάστημα της συσσωρευμένης πιθανότητας βρίσκεται και επιλέγεται το άτομο που ανήκει στο δεξιό άκρο του διαστήματος να περάσει στον νέο πληθυσμό. Άρα τα άτομα που θα περάσουν είναι τα 6, 2, 2, 8, 2, 3, 7 και 5. Παρατηρείται ότι το άτομο 4 με μηδενική απόδοση δεν συνεχίζει στον νέο πληθυσμό, όπως και το 1. Για κάθε άτομο που επιλέγεται δημιουργείται ένα ακριβές αντίγραφο και το σύνολο των αντιγράφων αποτελούν τον δοκιμαστικό πληθυσμό πάνω στον οποίο θα εφαρμοστούν οι παρακάτω γενετικές διαδικασίες.

Κατά τη διαδικασία της επιλογής ένα απευκταίο, αλλά όχι απίθανο ενδεχόμενο είναι να μην επιλεγεί το ικανότερο άτομο του πληθυσμού. Προκειμένου να αποφευχθεί κάτι τέτοιο εφαρμόζεται μια τεχνική που ονομάζεται *ελιτισμός* (elitism). Με τον ελιτισμό αντιγράφεται “χαριστικά” το ικανότερο άτομο στην επόμενη γενιά, προτού καν αρχίσει η επιλογή και εξασφαλίζεται έτσι η επιβίωσή του.

Η τεχνική του ελιτισμού ενέχει τον κίνδυνο της γρήγορης κυριαρχίας του πληθυσμού από ένα άτομο, το οποίο να μην έχει συνήθως μεγάλη ικανότητα, αλλά δεν αποτελεί προϊόν εξαντλητικού ψαξίματος και αποδοτικής εργασίας του αλγορίθμου και τις περισσότερες φορές δεν είναι το βέλτιστο σημείο του χώρου αναζήτησης. Γι' αυτόν ακριβώς το λόγο, και σύμφωνα με τη διδακτορική εργασία του De Jong (1975), ο ελιτισμός είναι προτιμότερο να εφαρμόζεται σε περιπτώσεις τοπικής αναζήτησης, όπου όντως μπορεί να βελτιώσει αξιοσημείωτα την απόδοση.

- **Διασταύρωση (Crossover):** Ο προσωρινός πληθυσμός που προέκυψε από τη προηγούμενη διαδικασία πρέπει να περάσει από τη διαδικασία ζευγαρώματος για να πραγματοποιηθεί ένα είδος γονιμοποίησης, όπως συμβαίνει και στη φύση. Η νέα, λοιπόν, ομάδα ατόμων που προέκυψε από την επιλογή σχηματίζει με τυχαίο τρόπο ομάδες των δύο. Το ποιος θα ζευγαρώσει με ποιον, από τα άτομα του προσωρινού πληθυσμού, ίσως να επηρεάζει την ταχύτητα σύγκλισης του αλγορίθμου. Όμως δεν έχει εντοπιστεί προς το παρόν βέλτιστος τρόπος ζευγαρώματος και σε όλες τις εφαρμογές το ζευγάρωμα γίνεται με τυχαίο τρόπο. Σε κάθε ομάδα, τα δύο μέλη παίρνουν μέρος σε μια απλή λειτουργία ανταλλαγής γενετικού υλικού που ονομάζεται διασταύρωση. Η διασταύρωση είναι μια απαραίτητη λειτουργία που συμβάλλει αποφασιστικά στην επίδοση ενός ΓΑ. Εξ αιτίας αυτής της σπουδαιότητας, έχει γίνει αρκετή έρευνα και έχουν επινοηθεί πολλοί τρόποι υλοποίησης του. Μερικοί μπορούν να εφαρμοστούν σε κάθε τύπο προβλήματος, ενώ άλλοι είναι πιο κατάλληλοι και εξειδικευμένοι για ειδικές περιπτώσεις. Στόχος της διασταύρωσης είναι η νέα γενιά που θα προκύψει μετά την εφαρμογή της να περιλαμβάνει άτομα που θα διαφέρουν από τους γονείς τους και θα φέρουν συνδυασμό των καλύτερων χαρακτηριστικών τους. Ερευνητές που ασχολούνται χρόνια με τους ΓΑ υποστηρίζουν ότι, αν αφαιρεθεί η διασταύρωση από έναν ΓΑ, τότε μειώνεται σημαντικά η απόδοσή του, αλλά αυτή δεν είναι μια άποψη με καθολική αποδοχή (Michalewicz, 1992).

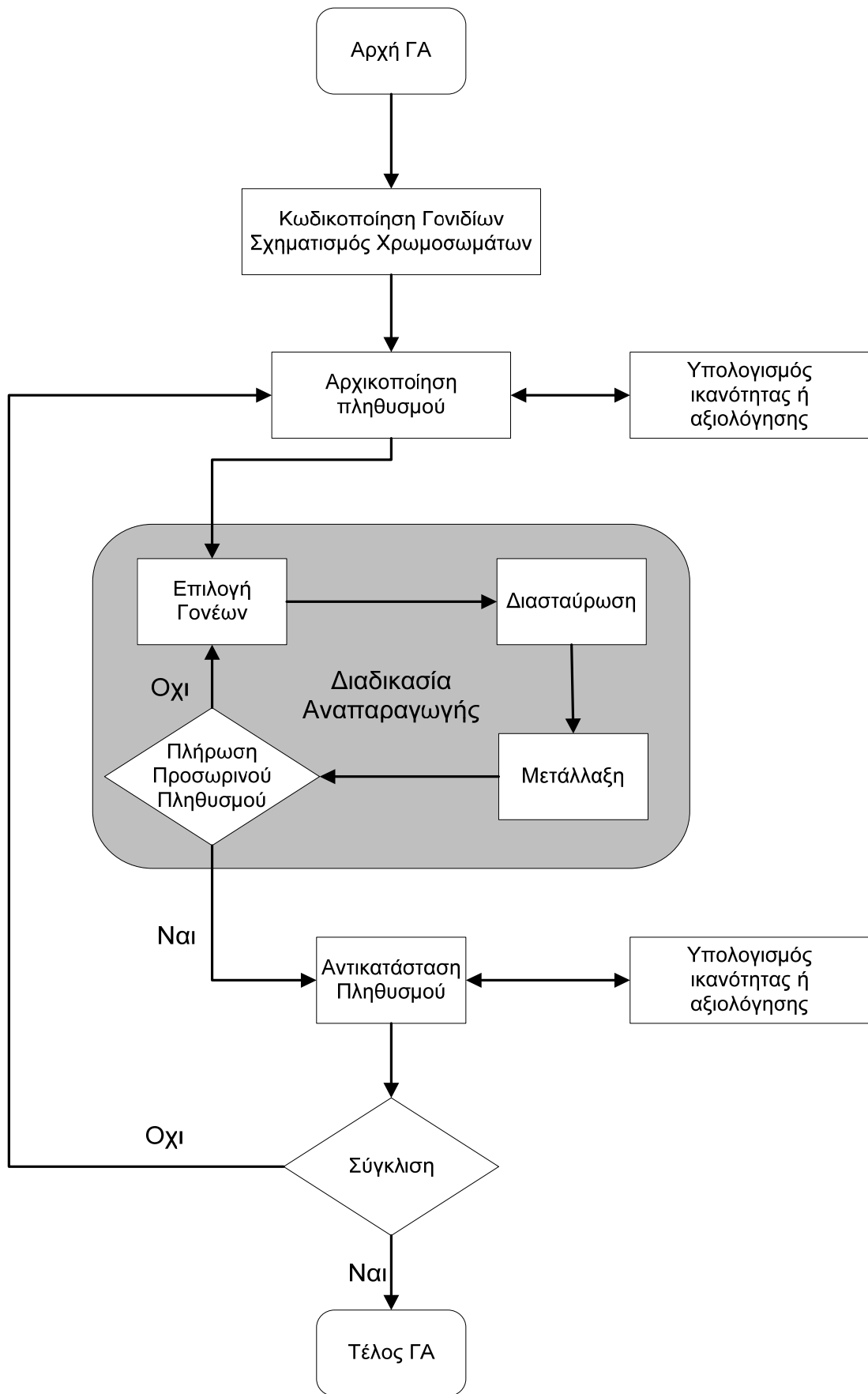
Ένα ενδεικτικό της χρησιμότητας της διασταύρωσης είναι η ανακατεύθυνση του ψαξίματος σε νέες περιοχές του χώρου αναζήτησης οι οποίες δεν έχουν διερευνηθεί ξανά. Έτσι διευρύνεται το πεδίο δράσης του αλγορίθμου και αυξάνουν οι πιθανότητες επιτυχίας του. Επίσης, τα νέα άτομα περιλαμβάνουν συνδυασμούς χαρακτηριστικών των γονέων τους και με αυτό τον τρόπο μπορούν να προκύψουν επιτυχημένοι συνδυασμοί υψηλής ικανότητας. Υπάρχει, βέβαια, το ενδεχόμενο η διασταύρωση να δώσει χειρότερα παιδιά από τους γονείς, αλλά αυτά δεν θα έχουν μεγάλη πιθανότητα πολλαπλασιασμού στον επόμενο αναπαραγωγικό κύκλο, λόγω μικρής απόδοσης. Στην πράξη, η διασταύρωση χρησιμοποιείται με παραμετροποιημένη μορφή, δηλαδή λαμβάνει χώρα με πιθανότητα, την λεγόμενη *πιθανότητα διασταύρωσης (crossover probability)* p_c , που καθορίζεται από το σχεδιαστή του ΓΑ. Συνήθως, αυτή η πιθανότητα ποικίλει από πρόβλημα σε πρόβλημα, ενώ είναι δυνατό και να αλλάζει κατά τον χρόνο τρεξίματος. Επίσης, πρέπει να αναφερθεί ότι η τιμή αυτής της πιθανότητας επηρεάζει το χρόνο τρεξίματος του αλγορίθμου, δηλαδή τη σύγκλισή του. Η τιμή $p_c=1$, σημαίνει συνεχή εφαρμογή του τελεστή διασταύρωσης, άρα το ψάξιμο γίνεται με μικρό βήμα. Αυτό θα έχει ως αποτέλεσμα η αναζήτηση να γίνει σε όλο το χώρο, άρα ο αλγόριθμος θα συγκλίνει στο βέλτιστο, αλλά πολύ αργά. Αντίθετα, χρησιμοποιώντας μικρές τιμές της p_c έχει σαν αποτέλεσμα το ψάξιμο να κάνει άλματα, άρα ο αλγόριθμος είναι πιθανόν να συγκλίνει πιο γρήγορα. Χρησιμοποιώντας μεγάλο βήμα, υπάρχει ο κίνδυνος, ο αλγόριθμος να ξεπεράσει το βέλτιστο και έτσι να αποκλίνει. Έτσι, επιλέγουμε συνήθως μεγάλο βήμα στην αρχή του ψαξίματος, και στη συνέχεια, όταν ο αλγόριθμος προσεγγίσει την τιμή του βέλτιστου, χρησιμοποιούμε μικρό βήμα αναζήτησης. Με αυτό τον τρόπο, μπορούμε να αυξήσουμε την ταχύτητα αναζήτησης, χωρίς να κινδυνεύουμε να αποκλίνει ο αλγόριθμος.

- **Μετάλλαξη (Mutation):** Μετά την διαδικασία της διασταύρωσης ακολουθεί η μετάλλαξη. Αν και είναι λιγότερη σημαντική, ωστόσο κρίνεται ως χρήσιμη. Είναι μια λειτουργία που όταν συμβαίνει αραιά στη φύση δρα βελτιωτικά για τους οργανισμούς και γενικά για την εξέλιξη της ζωής. Ανάλογος είναι ο ρόλος της και στα τεχνικά περιβάλλοντα. Η λειτουργία της είναι απλή: Ενεργεί σε ένα μόνο οργανισμό κάθε

φορά. Καθώς αντιγράφονται δυαδικά ψηφία από τον γονέα στον απόγονο, επιλέγεται τυχαία με μικρή πιθανότητα, τη λεγόμενη *πιθανότητα μετάλλαξης* (*mutation probability*) p_m , ένα ψηφίο και αντιστρέφεται (από 0 σε 1 ή το αντίστροφο), στην περίπτωση δυαδικής κωδικοποίησης. Αν έχει υπάρξει κάποιο άλλο είδος κωδικοποίησης, τότε η διαδικασία της μετάλλαξης μπορεί να εφαρμοστεί με την εύρεση π.χ. του συμπληρώματος ως προς το αριθμητικό σύστημα (δεκαδικό ή δεκαεξαδικό κ.ο.κ) της κωδικοποίησης, του ψηφίου που επιλέχτηκε με βάση τη πιθανότητα μετάλλαξης. Είναι πολύ σημαντικό η πιθανότητα να πραγματοποιηθεί η μετάλλαξη να είναι αρκετά μικρή (περίπου μία μετάλλαξη σε κάθε χίλια ψηφία που αντιγράφονται), γιατί σε αντίθετη περίπτωση ο ΓΑ εκφυλίζεται σε τυχαίο ψάξιμο.

Αν και υπάρχει κάποια σύγχυση για το ρόλο της μετάλλαξης, τόσο φυσικής όσο και τεχνητής, το σίγουρο είναι πως είναι απαραίτητη. Η μετάλλαξη λειτουργεί ως ασφαλιστική δικλείδα για τις περιπτώσεις, κατά τις οποίες η επιλογή και η διασταύρωση, ενδεχομένως, χάσουν κάποιες πολύτιμες γενετικές πληροφορίες. Όταν συμβαίνει, επιφέρει ποικιλία στον πληθυσμό, ανακατευθύνει την αναζήτηση και εξασφαλίζει ότι κανένα σημείο του χώρου αναζήτησης δεν μπορεί να αποκλειστεί εντελώς από τη διαδικασία του ψαξίματος (Λυκοθανάσης, 2001).

Προκειμένου να συνοψίσουμε όλα όσα αναφέρθηκαν σε αυτή την παράγραφο παρουσιάζεται το διάγραμμα ροής της εικόνας 2.2 για την λειτουργία ενός ΓΑ:



Εικόνα 2.2 - Διάγραμμα ροής της εφαρμογής του ΓΑ

2.2.6 Πλεονεκτήματα Γενετικών Αλγορίθμων

Η χρήση των ΓΑ σε διάφορες εφαρμογές είναι ελκυστική για αρκετούς λόγους (Λυκοθανάσης, 2001). Οι κυριότεροι, ίσως, είναι οι εξής :

1. *Μπορούν να λύσουν δύσκολα προβλήματα γρήγορα και αξιόπιστα:* Ένας από τους σημαντικούς λόγους χρήσης των ΓΑ είναι η μεγάλη τους αποδοτικότητα. Τόσο η θεωρία, όσο και η πράξη έχουν δείξει ότι προβλήματα που έχουν πολλές, δύσκολα προσδιορισμένες, λύσεις μπορούν να αντιμετωπιστούν καλύτερα από ΓΑ. Είναι δε αξιοσημείωτο ότι συναρτήσεις που παρουσιάζουν μεγάλες διακυμάνσεις και καθιστούν ανεπαρκείς άλλες μεθόδους στην εύρεση των ακρότατων τους, για τους ΓΑ αυτές οι διακυμάνσεις δεν αποτελούν σημεία δυσχέρειας.
2. *Μπορούν εύκολα να συνεργαστούν με τα υπάρχοντα μοντέλα και συστήματα:* Οι ΓΑ προσφέρουν το σημαντικό πλεονέκτημα της χρήσης τους με προσθετικό τρόπο στα μοντέλα που χρησιμοποιούνται σήμερα, μη απαιτώντας την επανασχεδίαση τους. Μπορούν εύκολα να συνεργαστούν με τον υπάρχοντα κώδικα, αποτελώντας ένα επιπλέον τμήμα του. Χρησιμοποιούν μόνο πληροφορίες της διαδικασίας ή συνάρτησης που πρόκειται να βελτιστοποιήσουν, δίχως να ενδιαφέρει άμεσα ο ρόλος της μέσα στο σύστημα ή η όλη δομή του συστήματος.
3. *Είναι εύκολα επεκτάσιμοι και εξελίξιμοι:* Οι ΓΑ δεν αντιστέκονται σε αλλαγές, επεκτάσεις και μετεξελίξεις, ανάλογα με την κρίση του σχεδιαστή. Σε πολλές εφαρμογές, έχουν αναφερθεί λειτουργίες των ΓΑ, που δεν είναι αντιγραμμένες από τη φύση ή που έχουν υποστεί σημαντικές αλλαγές, πάντα προς όφελος της απόδοσης. Παραλλαγές στο βασικό σχήμα δεν είναι απλά ανεκτές, αλλά σε ορισμένες περιπτώσεις επιβάλλονται.
4. *Μπορούν να συμμετέχουν σε υβριδικές μορφές με άλλες μεθόδους:* Αν και η ισχύς των ΓΑ είναι μεγάλη, σε μερικές ειδικές περιπτώσεις προβλημάτων, όπου άλλες μέθοδοι συμβαίνει να έχουν πολύ υψηλή αποδοτικότητα, λόγω εξειδίκευσης, υπάρχει η δυνατότητα χρησιμοποίησης ενός υβριδικού σχήματος ΓΑ με άλλη μέθοδο. Αυτό είναι αποτέλεσμα της μεγάλης ευελιξίας των ΓΑ.
5. *Εφαρμόζονται σε πολύ περισσότερα πεδία από κάθε άλλη μέθοδο:* Το χαρακτηριστικό, που τους εξασφαλίζει αυτό το πλεονέκτημα, είναι η ελευθερία επιλογής των κριτηρίων που καθορίζουν την επιλογή μέσα στο τεχνικό περιβάλλον. Έτσι, ΓΑ μπορούν να χρησιμοποιηθούν στην οικονομία, στο σχεδιασμό μηχανών, στην επίλυση μαθηματικών εξισώσεων, στην εκπαίδευση Νευρωνικών Δικτύων και σε πολλούς άλλους τομείς.
6. *Δεν απαιτούν περιορισμούς στις συναρτήσεις που επεξεργάζονται:* Ο κύριος λόγος που καθιστά τις παραδοσιακές μεθόδους δύσκαμπτες και ακατάλληλες για πολλά προβλήματα είναι η απαίτησή τους για ύπαρξη περιορισμών, όπως ύπαρξη παραγώγων, συνέχεια, όχι “θορυβώδεις” συναρτήσεις κτλ. Τέτοιου είδους ιδιότητες είναι αδιάφορες για τους ΓΑ πράγμα που τους κάνει κατάλληλους για μεγάλο φάσμα προβλημάτων.
7. *Δεν ενδιαφέρει η σημασία της υπό εξέταση πληροφορίας:* Η μόνη επικοινωνία του ΓΑ με το περιβάλλον του είναι η αντικειμενική συνάρτηση. Αυτό εγγυάται την επιτυχία του ανεξάρτητα από τη σημασία του προβλήματος. Βέβαια αυτό δε σημαίνει ότι δεν

υπάρχουν άλυτα προβλήματα για τους ΓΑ. Όπου όμως, δεν τα καταφέρνουν, η αιτία είναι η φύση του χώρου που ερευνούν και όχι το πληροφοριακό περιεχόμενο του προβλήματος.

8. *Έχουν από τη φύση τους το στοιχείο του παραλληλισμού:* Οι ΓΑ σε κάθε τους βήμα επεξεργάζονται μεγάλες ποσότητες πληροφορίας, αφού κάθε άτομο θεωρείται αντιπρόσωπος πολλών άλλων. Έχει υπολογιστεί ότι η αναλογία αυτή είναι της τάξεως $O(n^3)$, δηλαδή 10 άτομα αντιπροσωπεύουν περίπου 1000. Είναι, λοιπόν, προφανές ότι μπορούν να καλύψουν με αποδοτικό ψάξιμο μεγάλους χώρους σε μικρούς χρόνους.
9. *Είναι η μόνη μέθοδος που κάνει ταυτόχρονα εξερεύνηση του χώρου αναζήτησης και εκμετάλλευση της ήδη επεξεργασμένης πληροφορίας:* Ο συνδυασμός αυτός σπάνια συναντάται σε οποιαδήποτε άλλη μέθοδο. Με το τυχαίο ψάξιμο γίνεται καλή εξερεύνηση του χώρου, αλλά δεν γίνεται εκμετάλλευση της πληροφορίας. Αντίθετα, με την αναζήτηση με μικρά άλματα στη συνάρτηση (hill climbing) γίνεται καλή εκμετάλλευση της πληροφορίας, αλλά όχι καλή εξερεύνηση. Συνήθως τα δύο αυτά χαρακτηριστικά είναι ανταγωνιστικά και το επιθυμητό είναι να συνυπάρχουν και τα δύο προς όφελος της όλης διαδικασίας. Οι ΓΑ επιτυγχάνουν το βέλτιστο συνδυασμό εξερεύνησης και εκμετάλλευσης, πράγμα που τους κάνει ιδιαίτερα αποδοτικούς και ελκυστικούς.
10. *Επιδέχονται παράλληλη υλοποίηση:* Οι ΓΑ μπορούν να εκμεταλλευτούν τα πλεονεκτήματα των παράλληλων μηχανών, αφού λόγω της φύσης τους, εύκολα μπορούν να δεχτούν παράλληλη υλοποίηση. Το χαρακτηριστικό αυτό αυξάνει ακόμη περισσότερο την απόδοσή τους, ενώ σπάνια συναντάται σε ανταγωνιστικές μεθόδους.

2.2.7 Μειονεκτήματα Γενετικών Αλγορίθμων

Παρά την μεγάλη χρησιμότητα των ΓΑ σε πολλές εφαρμογές της καθημερινής ζωής, ωστόσο υπάρχουν κάποιοι λόγοι που θα μπορούσαν να σταθούν εμπόδιο στην ολοένα και μεγαλύτερη εξάπλωση αυτής της τεχνολογίας (Λυκοθανάσης, 2001). Συγκεκριμένα:

1. *Προβλήματα εξοικείωσης με τη Γενετική:* Γενικότερα οι έννοιες της Εξέλιξης και της Φυσικής Επιλογής μπορεί να μην ηχούν παράξενα σε αυτούς που ασχολούνται με την επιστήμη των υπολογιστών, αλλά δεν είναι και από τις πιο οικείες. Η Βιολογία δεν έχει άμεση σχέση με τους υπολογιστές, γι' αυτό και οι γνώσεις Βιολογίας σχεδόν όλων όσων ασχολούνται με τους υπολογιστές είναι σε πολύ γενικό επίπεδο. Παρόλα αυτά, για την κατανόηση των ΓΑ δεν απαιτούνται γνώσεις Γενετικής και Βιολογίας. Οι ΓΑ μιμούνται με αφαιρετικό τρόπο κάποιες διαδικασίες, που παρατηρούνται στη φύση, χωρίς να ενδιαφέρει σε μεγάλο βαθμό λεπτομέρειας η λειτουργία τους και χωρίς να είναι απαραίτητο το γνωστικό υπόβαθρο που έχουν οι βιολόγοι για να μελετήσουν αυτά τα φαινόμενα. Οι όροι είναι δανεισμένοι από τη βιολογία με σκοπό την καλύτερη εισαγωγή και κατανόηση του θέματος και όχι την παραπομπή του μελετητή στα άγνωστα πεδία μιας ξένης επιστήμης και, τελικά, τη σύγχυσή του. Θα μπορούσε, ίσως, να παραληφθεί η αναφορά στη Γενετική και να γίνει μια παρουσίαση των ΓΑ ως διαδικασίες για αναζήτηση και βελτιστοποίηση των μεταβλητών ενός προβλήματος, όμως, αυτό μάλλον θα έκανε τα πράγματα δυσκολότερα. Εξάλλου είναι συνηθισμένο το φαινόμενο θεωρίες που είναι δανεισμένες από άλλες επιστήμες να διατηρούν την αυθεντική τους ορολογία (π.χ. στα Νευρωνικά Δίκτυα: νευρώνες,

συνάψεις, κτλ.). Επιπλέον, η εξέλιξη των ΓΑ δεν είναι συνυφασμένη με την αντίστοιχη εξέλιξη των θεωριών της Βιολογίας. Το αρχικό μοντέλο είναι δανεισμένο από εκεί, όμως η εφαρμογή του στα Τεχνητά Συστήματα έγινε με πλήθος διαφοροποιήσεων και προσαρμοσέων, με στόχο πάντα τη βελτίωση της απόδοσης. Πλέον, μπορούμε να μιλάμε για εξέλιξη και απογόνους των πρώτων ΓΑ και για μια πορεία τους στο χρόνο, πορεία η οποία είναι πλήρως ανεξάρτητη και αυτοδύναμη.

2. Στη φύση, ως γνωστό, η εξέλιξη λειτουργεί με ρυθμούς πολύ αργούς. Χρειάζονται να περάσουν χιλιάδες γενιές, άρα και αρκετός χρόνος, για να αλλάξουν τα χαρακτηριστικά των ειδών και να διαφοροποιηθούν οι ικανότητες και η συμπεριφορά τους. Θέτουν, έτσι, ορισμένοι το ερώτημα: πώς είναι δυνατό ένα μοντέλο αναζήτησης λύσεων να έχει καλές επιδόσεις χρόνου, όταν είναι εμπνευσμένο από μια φυσική διαδικασία, που εξελίσσεται με ρυθμούς απίστευτα αργούς; Η απάντηση εδώ είναι απλή. Κατ' αρχήν, ακόμη και στη φύση, η εξέλιξη δεν είναι από μόνη της μια αργή διαδικασία. Εξέλιξη των ειδών συμβαίνει όταν αλλάζει το περιβάλλον τους και πρέπει να προσαρμοστούν στα καινούργια δεδομένα, ώστε να επιβιώσουν. Αλλαγές όμως, του περιβάλλοντος γίνονται με πολύ αργούς ρυθμούς και κατά συνέπεια και η εξέλιξη ακολουθεί αυτούς τους ρυθμούς. Αν οι αλλαγές του περιβάλλοντος γίνονται με γρηγορότερο τρόπο, τότε επιταχύνεται και η εξέλιξη. Αυτό, άλλωστε, παρατηρείται και στα βιολογικά εργαστήρια, όπου μικροοργανισμοί αλλάζουν τη συμπεριφορά τους αμέσως, όταν τοποθετούνται σε νέες συνθήκες. Επιπλέον, στο πεδίο των υπολογιστών, τα άτομα κωδικοποιούνται συνήθως ως συμβολοσειρές και οι συνθήκες του περιβάλλοντος μοντελοποιούνται με απλές μαθηματικές σχέσεις. Έτσι, το μοντέλο με το οποίο δουλεύει ο υπολογιστής δεν παρουσιάζει ιδιαίτερο υπολογιστικό φόρτο, συγκρινόμενο πάντα με αντίστοιχες μεθόδους. Το πλήθος των ατόμων, που κάθε φορά εξετάζεται, είναι από λίγες δεκάδες έως μερικές χιλιάδες, δηλαδή αρκετές τάξεις μεθόδους κάτω από το πλήθος των γονιδίων των χρωμοσωμάτων ενός έμβριου όντος. Ο ρυθμός που μπορούν να ζευγαρώνουν τα άτομα στους πιο γρήγορους υπολογιστές μπορεί να φτάσει το ένα εκατομμύριο ανά δευτερόλεπτο. Όσο μεγάλος και αν είναι ο χώρος που καλείται ο αλγόριθμος να ψάξει, η επεξεργασία μερικών μόνο ατόμων αρκεί, γιατί, όπως θα αναπτυχθεί και παρακάτω, τα άτομα αυτά θεωρούνται αντιπρόσωποι ολόκληρων κλάσεων. Έτσι, λοιπόν, οι ταχύτητες που μπορούν να επιτύχουν οι ΓΑ είναι πολύ υψηλές. Επίσης, όπως θα δούμε σε επόμενα, το μήκος της γενιάς επηρεάζει σημαντικά την ταχύτητα της εξέλιξης.

2.3 Παρουσίαση της Βιβλιοθήκης Γενετικών Αλγορίθμων Pyevolve στην Γλώσσα Προγραμματισμού Python

2.3.1 Γενικά

Η γλώσσα προγραμματισμού python διαθέτει μια σειρά από βιβλιοθήκες, μέσω των οποίων επιτυγχάνεται η επίλυση προβλημάτων με χρήση ΓΑ. Τρεις βιβλιοθήκες που υλοποιούν την χρήση αυτή είναι οι pyevolve, inspyred και pygene. Από αυτές τις τρεις, στις διαδικασίες που υλοποιούνται στα κεφάλαια που ακολουθούν, χρησιμοποιήθηκε η pyevolve v0.6 rc1 καθώς ήταν η πληρέστερη όσον αφορά τις διαδικασίες ενός ΓΑ καθώς και η πιο πρόσφατα αναθεωρημένη.

Καθότι το πρόβλημα που εξετάζεται χαρακτηρίζεται ως σύνθεση προβλήματος τηλεπισκόπησης σε συνδυασμό με ΓΑ, θα εξεταστούν στα εδάφια παρακάτω οι επιλογές - δυνατότητες της βιβλιοθήκης pyevolve μέσω συγγραφής κώδικα.

2.3.2 Οργάνωση ενός Προβλήματος

Ένας ΓΑ στην γλώσσα rython με χρήση της συγκεκριμένης βιβλιοθήκης αναζητά είτε το ελάχιστο είτε το μέγιστο μιας δοθείσας συνάρτησης. Όσο πιο ευρύ είναι το πεδίο αναζήτησης τόσοι περισσότερες πιθανότητες υπάρχουν να μην αποκλειστεί σε κάποιο τοπικό ελάχιστο ή μέγιστο αλλά να εντοπίσει το ολικό ελάχιστο ή μέγιστο αντίστοιχα.

Προκειμένου να λειτουργήσει σωστά ο ΓΑ πρέπει να υλοποιηθούν οι παρακάτω βασικές συνθήκες:

- Σωστή μελέτη του προβλήματος που πρέπει να επιλυθεί,
- Δημιουργία, με βάση το πρόβλημα, της συνάρτησης αξιολόγησης η οποία αποτελεί και τον θεμέλιο λίθο για την υλοποίηση του ΓΑ. Εάν η συνάρτηση δεν υλοποιηθεί σωστά τότε τα αποτελέσματα δεν θα είναι τα αναμενόμενα και το πρόβλημα δεν θα επιλυθεί ορθά,
- Καθορισμός του πλήθους των παραμέτρων που θα ελεγχθούν από τον ΓΑ και κατά συνέπεια δημιουργία ενός στιγμιότυπου χρωμοσώματος με βάση το πλήθος των παραμέτρων,
- Ορισμός των ορίων των παραμέτρων του ΓΑ και
- Υλοποίηση του ΓΑ

Ένα απλό παράδειγμα αποτελεί το πρόβλημα της εύρεσης ενός μονοδιάστατου πίνακα με 20 στοιχεία ο οποίος να έχει σε όλες του τις θέσεις τιμή μηδέν. Ως συνάρτηση αξιολόγησης μπορεί να υλοποιηθεί το άθροισμα των μηδενικών στοιχείων του πίνακα με αποτέλεσμα επιδίωξη του ΓΑ να αποτελεί η μεγιστοποίηση της συνάρτησης αξιολόγησης.

Ο κώδικας που υλοποιείται για την συνάρτηση αξιολόγησης είναι:

```
def eval_func(chromosome):
    score = 0.0

    # iterate over the chromosome elements (items)
    for value in chromosome:
        if value==0:
            score += 1.0

    return score
```

Ακολουθεί η δημιουργία ενός στιγμιότυπου χρωμοσώματος το οποίο αποτελείται από το πλήθος των παραμέτρων δηλαδή 20. Ο κώδικας είναι:

```
genome = G1DList.G1DList(20)
```

Η συνάρτηση αξιολόγησης πρέπει να ενσωματωθεί στον ΓΑ. Αυτό υλοποιείται μέσω της γραμμής κώδικα:

```
genome.evaluator.set(eval_func)
```

Επόμενο βήμα για την υλοποίηση του ΓΑ αποτελεί ο καθορισμός των ορίων των τιμών για τις 20 παραμέτρους, όπου για το συγκεκριμένο πρόβλημα το σύνολο των παραμέτρων μπορεί να κυμανθεί ανάμεσα στις τιμές 0 -10. Αυτό υλοποιείται με τον εξής κώδικα:

```
genome.setParams(rangemin=0, rangemax=10)
```

Τέλος ακολουθεί η δημιουργία της μηχανής του ΓΑ η οποία θα υλοποιήσει τον ΓΑ και η υλοποίηση της εξέλιξης των γενεών. Αυτά υλοποιούνται με τις γραμμές κώδικα αντίστοιχα:

```
ga = GSimpleGA.GSimpleGA(genome)
```

```
ga.evolve(freq_stats=10)
```

Το αποτέλεσμα του ΓΑ αποτελεί η λίστα

```
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```

Όπου σε κάθε στοιχείο βρίσκεται η τιμή 0 και η συνάρτηση αξιολόγησης έχει τιμή 20.

Προκειμένου όμως να δώσει ο ΓΑ το παραπάνω αποτέλεσμα υπάρχει μια σειρά από επιλογές οι οποίες περνούν στον ΓΑ διάφορα ορίσματα, ώστε αυτός να μπορέσει να δώσει αποτέλεσμα. Καθώς το πλήθος των επιλογών είναι μεγάλο, και κάποιες από αυτές είναι εντελώς προαιρετικές, παρουσιάζονται οι κυριότερες από αυτές:

- ✓ Initializers, δημιουργία αρχικού πληθυσμού,
- ✓ Population size, μέγεθος πληθυσμού ανά γενιά
- ✓ Selection Methods, μέθοδος επιλογής των ατόμων ανά γενιά,
- ✓ Set Elitism, ορισμός ατόμων με την καλύτερη απόδοση ανά γενιά,
- ✓ Crossover Methods, μέθοδος διασταύρωσης επιλεγμένων ατόμων,
- ✓ Crossover Rate, ποσοστό διασταύρωσης,
- ✓ Mutator Methods, μέθοδος μετάλλαξης επιλεγμένων ατόμων,
- ✓ Mutator Rate, ποσοστό μετάλλαξης
- ✓ Migration, μέθοδος μετανάστευσης ατόμων από γενιά σε γενιά
- ✓ Συνθήκες τερματισμού ενός ΓΑ (Generations, Converge και FitnessStats),
- ✓ Fitness Scaling Methods, μέθοδοι κλιμάκωσης των αποτελεσμάτων της συνάρτησης αξιολόγησης,
- ✓ DBAdapters, δημιουργία βάσης με στατιστικά αποτελέσματα από την εκτέλεση του ΓΑ και
- ✓ Graphical Plotting Tool Options για απεικόνιση διαγραμμάτων με βάση τα στατιστικά αποτελέσματα της παραπάνω βάσης

Στα εδάφια 2.3.4 έως 2.3.16 αναλύονται οι διαφορετικές δυνατότητες επιλογής τιμών για κάθε μια από τις παραπάνω επιλογές.

2.3.3 Αναπαραστάσεις ενός Χρωμοσώματος

Μέσω της συγκεκριμένης βιβλιοθήκης ένα χρωμόσωμα είναι δυνατό να αναπαρασταθεί με πέντε διαφορετικούς τρόπους. Αυτοί είναι οι εξής:

- ❖ Μονοδιάστατη λίστα (1D List)
- ❖ Δισδιάστατη λίστα (2D List)
- ❖ Μονοδιάστατη δυαδική λίστα (1D Binary List)
- ❖ Δισδιάστατη δυαδική λίστα (2D Binary List)
- ❖ Δέντρο

Οι επιλογές της μονοδιάστατης και δισδιάστατης λίστας καθώς και του δέντρου μπορούν να περιέχουν οποιαδήποτε μορφής αντικείμενα της `python`.

Καθώς στο πρόβλημα που αναπτύσσεται στην συγκεκριμένη διπλωματική χρησιμοποιείται η μονοδιάστατη λίστα, θα αναλυθούν παρακάτω οι ιδιότητες που αναφέρονται σε αυτήν. Ορισμένες από τις ιδιότητες αυτής ισχύουν και στις άλλες μορφές,

ωστόσο υπάρχουν και ιδιότητες οι οποίες αφορούν αποκλειστικά κάποια ή κάποιες από τις άλλες μορφές. Αυτές δεν εξετάζονται παρακάτω.

2.3.4 Ανάλυση της επιλογής Initializers

Ο αρχικός πληθυσμός δημιουργείται με την βοήθεια μιας ομοιόμορφης τυχαίας γεννήτριας αριθμών. Ορίζεται ένα συγκεκριμένο εύρος τιμών είτε συνολικά για όλες τις παραμέτρους είτε ατομικά για την κάθε μία. Με αυτό τον τρόπο όλες οι τιμές του αρχικού πληθυσμού βρίσκονται μέσα σε αυτό το εύρος. Για παράδειγμα ένας πληθυσμός μεγέθους 3 σε ένα πρόβλημα με 2 μεταβλητές και εύρος τιμών 1 - 10 θα είναι όπως παρακάτω:

```
2 3
5 1
7 9
```

Οι διαθέσιμες επιλογές για την αρχικοποίηση είναι οι παρακάτω:

- G1DListInitializerInteger: Αρχικοποίηση του πληθυσμού με ακέραιες τιμές. Αποτελεί την προεπιλεγμένη επιλογή.
- G1DListInitializerReal: Αρχικοποίηση του πληθυσμού με πραγματικές τιμές. Η επιλογή αυτή και η προηγούμενη χρησιμοποιούνται στην περίπτωση που το εύρος ορίων των τιμών είναι κοινό για όλες τις παραμέτρους.
- G1DListInitializerAllele: Η επιλογή αυτή χρησιμοποιείται σε περίπτωση που κάθε παράμετρος ή κάποιες από τις παραμέτρους έχουν διαφορετικά όρια τιμών. Στην περίπτωση αυτή υπάρχει όρισμα στην επιλογή που καθορίζει το εύρος ορίων τιμών των παραμέτρων το οποίο λαμβάνει τιμή True ένα πρόκειται να χρησιμοποιηθούν πραγματικοί αριθμοί και τιμή False στην αντίθετη περίπτωση. Για παράδειγμα η αρχικοποίηση με ακέραιες τιμές μιας παραμέτρου υλοποιείται με τον κώδικα

```
GAllele.GAlleleRange(1, 255, real=False)
```

2.3.5 Ανάλυση της Επιλογής Population Size

Μέσω της επιλογής Population size καθορίζεται ο αριθμός των ατόμων που υπάρχουν σε κάθε γενιά. Εάν ο πληθυσμός κάθε γενιάς είναι μεγάλος σε αριθμό τότε ο ΓΑ αναζητάει βέλτιστη λύση με μεγαλύτερη λεπτομέρεια, μειώνοντας τις πιθανότητες εύρεσης ενός τοπικού αντί για ολικού ελαχίστου ή μεγίστου. Ωστόσο στην περίπτωση αυτή αυξάνει ο χρόνος εκτέλεσης του αλγορίθμου.

Το προκαθορισμένο μέγεθος πληθυσμού είναι 80 άτομα ανά γενιά. Σε προβλήματα με μεγάλο πλήθος μεταβλητών, το μέγεθος αυτό μπορεί να μην είναι αρκετό. Συνήθως απαιτείται μεγαλύτερο μέγεθος πληθυσμού για τέτοια προβλήματα. Μικρότερος πληθυσμός απαιτείται για προβλήματα με μικρότερο αριθμό μεταβλητών. Εάν ένα πρόβλημα αποτελείται από 2 μεταβλητές, ένας πληθυσμός 10 ατόμων κρίνεται ικανοποιητικός.

Η επιλογή αυτή π.χ. για μέγεθος πληθυσμού 40 ατόμων, γίνεται με την εφαρμογή της εντολής:

```
ga.setPopulationSize(40)
```

Ένας από τους σοβαρότερους παράγοντες που καθορίζει την απόδοση της εκτέλεσης του ΓΑ είναι η πυκνότητα (ποικιλομορφία) του πληθυσμού. Εάν η μέση απόσταση μεταξύ των ατόμων είναι μεγάλη, η ποικιλομορφία είναι υψηλή ενώ εάν η μέση απόσταση είναι μικρή, η ποικιλομορφία είναι χαμηλή. Η σωστή επιλογή της πυκνότητας του πληθυσμού είναι κυρίως θέμα δοκιμής. Εάν η ποικιλομορφία είναι πάρα πολύ υψηλή ή πάρα πολύ χαμηλή, είναι πολύ πιθανό ο αλγόριθμος να μην αποδώσει σωστά.

2.3.6 Ανάλυση της Επιλογής Selection Methods

Η επιλογή αυτή παρέχει την δυνατότητα προσδιορισμού του τρόπου, με τον οποίο ο ΓΑ θα επιλέξει από έναν πληθυσμό, τα άτομα που θα αποτελέσουν τους γονείς για την δημιουργία της επόμενης γενιάς. Διαθέσιμες συναρτήσεις για την εφαρμογή της επιλογής αυτής είναι:

- **Rank**, με βάση την σειρά κατάταξης: Με την επιλογή αυτή επιλέγονται τα καλύτερα άτομα του πληθυσμού σε κάθε γενιά ώστε να αποτελέσουν τους γονείς και να εφαρμοστούν περαιτέρω διεργασίες σε αυτά. Αποτελεί την προεπιλεγμένη επιλογή.
- **Roulette Wheel**, ρουλέτα: Η επιλογή των γονέων γίνεται με την προσομοίωση μιας ρουλέτας - τροχού, στην οποία η έκταση του τμήματος του τροχού που αντιστοιχεί σε κάθε άτομο είναι ανάλογη της τιμής προσδοκίας του ατόμου. Ο αλγόριθμος χρησιμοποιεί ένα τυχαίο αριθμό για να επιλέξει ένα από τα παραπάνω τμήματα, και να χρησιμοποιήσει το άτομο του τμήματος αυτού ως γονέα
- **Tournament**, τουρνουά ομάδων: Επιλέγονται τυχαία άτομα με βάση την διαδικασία της ρουλέτας, που συγκροτούν μία ομάδα (το πλήθος των ατόμων είναι συνάρτηση του πληθυσμού κάθε γενιάς). Γονέας είναι το άτομο με την καλύτερη απόδοση στην κάθε ομάδα.
- **Alternative Tournament**, εναλλακτικό τουρνουά ομάδων : Στην περίπτωση αυτή ο χρήστης ορίζει το πλήθος της κάθε ομάδας, ενώ δεν χρησιμοποιείται η διαδικασία της ρουλέτας για την επιλογή των ατόμων αλλά τυχαία διαδικασία.
- **Uniform**, διάσπαρτη: Η επιλογή των γονέων γίνεται ανάλογα με τις προσδοκίες και τον αριθμό των γονέων που απαιτούνται. Η επιλογή αυτή είναι χρήσιμη για τον εντοπισμό σφαλμάτων και τις δοκιμές, αλλά δεν αποτελεί αποτελεσματική στρατηγική αναζήτησης.
- **key_fitness_score**, τιμή συνάρτησης αξιολόγησης: Η επιλογή των γονέων γίνεται με βάση την καλύτερη τιμή της καταλληλότητας κάθε ατόμου, δηλαδή της αρχικής τιμής της συνάρτησης αξιολόγησης σε σχέση με τα υπόλοιπα άτομα του πληθυσμού σε κάθε γενιά.
- **key_raw_score**, αρχική τιμή συνάρτησης καταλληλότητας: Η επιλογή των γονέων γίνεται με βάση την καλύτερη αρχική τιμή της συνάρτησης αξιολόγησης κάθε ατόμου.

Η σύνταξη που πρέπει να έχει η συνάρτηση για παράδειγμα στην επιλογή **Tournament** πρέπει να είναι η ακόλουθη:

```
ga.selector.set(Selectors.GTournamentSelector)
```

2.3.7 Ανάλυση της Επιλογής setElitism και setElitism Replacement

Η επιλογή **setElitism**, ορίζει εάν άτομα με την καλύτερη απόδοση σε κάθε γενιά θα επιβιώσουν στην επόμενη. Τα άτομα που δημιουργούνται μέσω της επιλογής αυτής ονομάζονται **elite children**. Εάν οριστεί True τότε η προκαθορισμένη τιμή για αυτήν την επιλογή είναι 1 άτομο ανά γενιά. Προκειμένου να τροποποιηθεί η τιμή αυτή χρησιμοποιείται η εντολή **setElitismReplacement**.

Όταν η τιμή που δίνεται είναι τουλάχιστον 1, τότε η τιμή της συνάρτησης αξιολόγησης μπορεί μόνο να μειωθεί από γενιά σε γενιά, γεγονός που αποτελεί και την επιθυμητή λειτουργία του αλγορίθμου έως ότου ελαχιστοποιηθεί η συνάρτηση αξιολόγησης. Μεγάλες τιμές στην επιλογή αυτή οδηγεί στο γεγονός τα καταλληλότερα άτομα να πρωταγωνιστούν στον πληθυσμό κάθε γενιάς, με αποτέλεσμα την πιθανή, λιγότερο αποτελεσματική αναζήτηση του αλγορίθμου, για νέα ικανότερα άτομα.

Για παράδειγμα εάν το μέγεθος του πληθυσμού είναι 20 και η επιλογή **Elite count** είναι 2, αυτό σημαίνει ότι τα 18 άτομα κάθε γενιάς με τις χειρότερες αποδόσεις θα χρησιμοποιηθούν στον ΓΑ για την διασταύρωση και την μετάλλαξη, ώστε να προκύψουν σε κάθε επόμενη γενιά νέα τέκνα. Η επιλογή αυτή π.χ. για τιμή 4, γίνεται με την εφαρμογή της εντολής:

```
ga.setElitismReplacement(4)
```

2.3.8 Ανάλυση της Επιλογής Crossover Methods

Η επιλογή αυτή καθορίζει τον τρόπο διασταύρωσης των ατόμων μιας γενιάς ώστε να προκύψουν οι απόγονοι της επόμενης γενιάς. Διαθέσιμες συναρτήσεις για την εφαρμογή της επιλογής αυτής είναι:

- **Single Point**, μονού σημείου: Επιλέγεται στη τύχη ένας ακέραιος n από το 1 έως και το πλήθος των μεταβλητών. Στον απόγονο τα γονίδια 1 έως n προέρχονται από τον πρώτο γονέα και τα γονίδια $n+1$ έως τέλος από τον άλλο γονέα. Αποτελεί την προεπιλεγμένη επιλογή. Για παράδειγμα, εάν $p1=[a b c d e f g h]$, $p2=[1 2 3 4 5 6 7 8]$ οι γονείς και $n=3$ οπότε $child1=[a b c 4 5 6 7 8]$ και $child2=[1 2 3 d e f g h]$.



Εικόνα 2.3 - Διασταύρωση μονού σημείου
Πηγή: wikipedia

- **Two point**, διπλού σημείου: Πρόκειται για την εφαρμογή της διασταύρωσης μονού σημείου δύο φορές, αφού πρώτα επιλεγούν με τυχαίο τρόπο δύο θέσεις κοπής από το 1 έως το πλήθος των μεταβλητών. Για παράδειγμα, εάν $p1=[a b c d e f g h]$, $p2=[1 2 3 4 5 6 7 8]$ οι γονείς και $n=3$, $m=6$ τότε $child1=[a b c 4 5 6 g h]$ και $child2=[1 2 3 d e f 7 8]$.



Εικόνα 2.4 - Διασταύρωση διπλού σημείου
Πηγή: wikipedia

- **Uniform**, ομοιόμορφη: Στην περίπτωση αυτή χρησιμοποιείται μια σταθερή αναλογία αναμίξεως μεταξύ των δυο γονέων. Σε αντίθεση με τις δυο προηγούμενες μεθόδους, η συγκεκριμένη επιτρέπει την χρησιμοποίηση επιπέδου γονιδίου από κάθε γονέα αντί για επίπεδο τμήματος. Η μέθοδος αυτή αποτελεί μια πιο διερευνητική μέθοδο σε σχέση με τις προηγούμενες. Το αποτέλεσμα αποτελεί μια πιο ολοκληρωμένη έρευνα του χώρου σχεδιασμού. Για παράδειγμα, εάν $p1 = [a b c d e f g h]$ και $p2 = [1 2 3 4 5 6 7 8]$ είναι οι γονείς τότε τα παιδιά θα είναι: $child1 = [a 2 c 4 e f 7 8]$ και $child2 = [1 b 3 d 5 6 g h]$.



Εικόνα 2.5 - Διασταύρωση ομοιόμορφη
Πηγή: wikipedia

- **Cut and CrossFill**, αποκοπή και γέμισμα: Στην περίπτωση αυτή αρχικά επιλέγεται ένα τυχαίο σημείο ανάμεσα στις διαθέσιμες θέσεις, ανάλογα με το πλήθος των μεταβλητών. Εν συνεχεία γίνεται αποκοπή σε δυο τμήματα κάθε γονέα με βάση το σημείο αυτό. Επιλέγεται το πρώτο τμήμα από κάθε γονέα να αποτελέσει το πρώτο τμήμα για κάθε παιδί. Τέλος κάθε παιδί συμπληρώνεται με τιμές που υπάρχουν στον άλλο σε αριθμηση γονέα (δηλαδή το πρώτο παιδί με τιμές του δεύτερου γονέα) και οι οποίες δεν υφίστανται στο πρώτο τμήμα του παιδιού. Για παράδειγμα εάν ο πρώτος γονέας είναι $p_1 = [1\ 3\ 5\ | \ 7\ 6\ 2\ 4\ 8]$ και $p_2 = [2\ 1\ 8\ | \ 6\ 4\ 3\ 5\ 7]$ και έχει επιλεγεί ως σημείο διασταύρωσης, όπως φαίνεται, η τρίτη θέση τότε το πρώτο τμήμα του πρώτου παιδιού θα είναι $child_1 = [1\ 3\ 5\ \dots]$ και του δεύτερου παιδιού $child_2 = [2\ 1\ 8\ \dots]$. Για να συμπληρωθεί το πρώτο παιδί επιλέγονται από τον δεύτερο γονέα τιμές που δεν υπάρχουν στο πρώτο τμήμα του παιδιού οπότε τελικά $child_1 = [1\ 3\ 5\ 2\ 8\ 6\ 4\ 7]$ και αντίστοιχα συμβαίνει για το δεύτερο παιδί οπότε $child_2 = [2\ 1\ 8\ 3\ 5\ 7\ 6\ 4]$.
- **Edge**, γεινίαση: Η μέθοδος αυτή βασίζεται στην ιδέα ότι οι απόγονοι πρέπει να δημιουργούνται μόνο από τους υπάρχοντες γείτονες των γονιδίων του κάθε γονέα. Ο αλγόριθμος λειτουργίας της μεθόδου αυτής είναι:
 - Δημιουργία του πίνακα γεινίασης για κάθε γονίδιο,
 - Επιλογή ενός τυχαίου γονιδίου και τοποθέτηση του ως πρώτο στοιχείο του απόγονου,
 - Εξέταση των γειτόνων του στοιχείου αυτού. Εάν υπάρχει κοινός γείτονας παραπάνω από μια φορά τότε επιλέγεται αυτός. Σε περίπτωση που δεν ισχύει το προηγούμενο τότε επιλέγεται ο γείτονας που έχει τους λιγότερους γείτονες. Το κάθε στοιχείο που επιλέγεται κάθε φορά αφαιρείται από τις λίστες του πίνακα γειτόνων. Εάν το αποτέλεσμα είναι κενό ή όλα τα στοιχεία έχουν ίδιο αριθμό γειτόνων τότε γίνεται και πάλι τυχαία επιλογή.

Έστω ότι υπάρχουν δυο γονείς $p_1 = [1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9]$ και $p_2 = [9\ 3\ 7\ 8\ 2\ 6\ 5\ 1\ 4]$. Αρχικά δημιουργείται ο πίνακας γεινίασης δηλαδή

Γονίδιο	Γείτονες	Γονίδιο	Γείτονες
1	2,4,5,9	6	2,5,5,7
2	1,3,6,8	7	3,6,8,8
3	2,4,7,9	8	2,7,7,9
4	1,3,5,9	9	1,3,4,8
5	1,4,6,6		

Πίνακας 2.2 - Πίνακας γειτόνων κάθε γονιδίου στο χρωμόσωμα

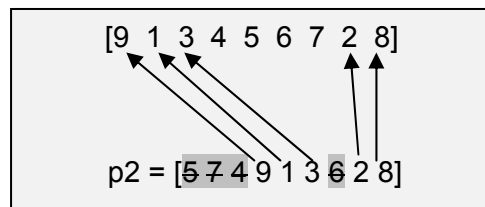
Εν συνεχεία επιλέγεται τυχαία έστω το γονίδιο 1. Επειδή δεν υπάρχει κοινός γείτονας σε αυτό επιλέγεται το στοιχείο με τους λιγότερους γείτονες δηλαδή το 5. Το στοιχείο 5 έχει κοινό γείτονα το στοιχείο 6, το οποίο και αποτελεί το επόμενο στοιχείο του απογόνου. Οι γείτονες του (2,7) έχουν ίδιο αριθμό γειτόνων άρα γίνεται τυχαία

επιλογή. Επιλέγεται το **2**. Από τους γείτονες του (3,8) το **8** έχει τους λιγότερους γείτονες. Το 8 έχει κοινό γείτονα το στοιχείο **7**, που επιλέγεται. Το 7 έχει το μόνο διαθέσιμο γείτονα το **3**, όπου και επιλέγεται. Το 3 έχει γείτονες τα 4,9 όπου και τα δυο στοιχεία έχουν πλέον από ένα γείτονα οπότε με τυχαία επιλογή επιλέγεται το **9**. Το τελευταίο διαθέσιμο στοιχείο είναι το **4**. Άρα το παιδί θα είναι [1,5,6,2,8,7,3,9,4]. Στον πίνακα 2.3 παρουσιάζεται η διαδικασία που αναφέρθηκε. Παρατηρείται ότι με την διαδικασία αυτή δημιουργείται ένας απόγονος από δυο γονείς.

Επιλογή	Επιλεγμένο Στοιχείο	Λόγος επιλογής	Αποτέλεσμα
Όλα	1	Τυχαία επιλογή	[1]
2,5,4,9	5	Λιγότεροι γείτονες	[1,5]
4,6	6	Κοινός γείτονας	[1,5,6]
2,7	2	Τυχαία επιλογή λόγω κοινού αριθμού γειτόνων	[1,5,6,2]
3,8	8	Λιγότεροι γείτονες	[1,5,6,2,8]
7,9	7	Κοινός γείτονας	[1,5,6,2,8,7]
3	3	Μοναδικό στοιχείο	[1,5,6,2,8,7,3]
4,9	9	Τυχαία επιλογή λόγω κοινού αριθμού γειτόνων	[1,5,6,2,8,7,3,9]
4	4	Τελευταίο στοιχείο	[1,5,6,2,8,7,3,9,4]

Πίνακας 2.3 - Διαδικασία υλοποίησης της μεθόδου διασταύρωσης γεινίασης

- **Order (OX)**, διατεταγμένη: Στην μέθοδο αυτή επιλέγεται αρχικά τυχαία ένα υποσύνολο του αρχικού χρωμοσώματος του πρώτου γονέα. Το υποσύνολο αυτό τοποθετείται στην ανάλογη θέση με την θέση που κατείχε ώστε να δημιουργηθεί ο απόγονος. Διαγράφονται τα γονίδια που περιέχονται στο υποσύνολο από τον δεύτερο γονέα. Τέλος τοποθετούνται στις υπόλοιπες θέσεις του απογόνου, με διατεταγμένη σειρά, τα υπόλοιπα γονίδια του δεύτερου γονέα. Στην μέθοδο αυτή όπως και στην μέθοδο γεινίασης δημιουργείται ένας απόγονος από δυο γονείς. Για παράδειγμα θεωρούνται οι γονείς $p1 = [1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9]$ και $p2 = [5\ 7\ 4\ 9\ 1\ 3\ 6\ 2\ 8]$. Επιλέγονται τυχαία από τον πρώτο γονέα τα γονίδια [4 5 6 7]. Οπότε ο απόγονος δημιουργείται ως εξής:



Η σύνταξη που πρέπει να έχει η συνάρτηση για παράδειγμα στην επιλογή **OX** πρέπει να είναι η ακόλουθη:

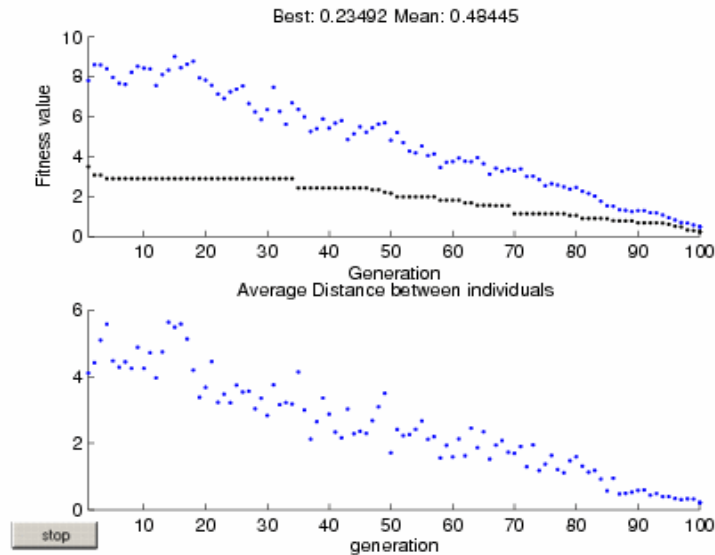
```
genome.crossover.set(Crossovers.G1DListCrossoverOX)
```

2.3.9 Ανάλυση της Επιλογής Crossover Rate

Η επιλογή αυτή, καθορίζει το ποσοστό του εκάστοτε πληθυσμού, εκτός από τα παιδιά που περιλαμβάνονται στην επιλογή **Elite**, το οποίο ουσιαστικά προέρχεται από διασταύρωση. Η προκαθορισμένη τιμή για αυτήν την επιλογή είναι 0.9 δηλαδή 90%. Η καλύτερη επιλογή

κυμαίνεται από 80% - 95%. Τιμή ίση με ένα σημαίνει ότι όλα τα παιδιά εκτός από τα άτομα ελίτ είναι παιδιά διασταυρώσεων, ενώ τιμή ίση με μηδέν σημαίνει ότι όλα τα παιδιά είναι παιδιά μετάλλαξης. Στην πραγματικότητα καμία από τις δυο παραπάνω επιλογές δεν αποτελεί αποτελεσματική στρατηγική. Για την ισχυροποίηση αυτής της άποψης παρατίθεται το παρακάτω παράδειγμα:

Έστω μια συνάρτηση αξιολόγησης $f(x)=|x_1|+|x_2|+\dots+|x_n|$ όπου $n=10$, της οποίας αναζητείται η εύρεση της ελάχιστης τιμής μέσω εφαρμογής ΓΑ. Αρχικά δίνεται τιμή Crossover Rate ίση με 0.8, οπότε με εφαρμογή του ΓΑ προκύπτουν τα εξής:

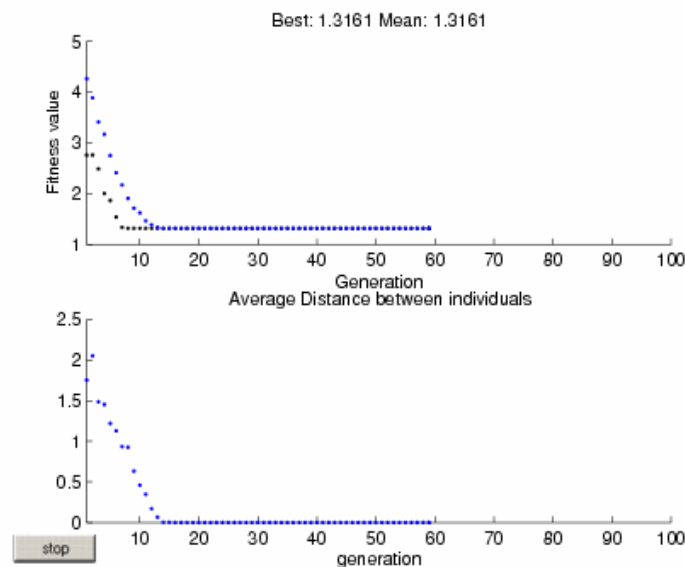


Εικόνα 2.6 - Διάγραμμα καλύτερης και μέσης τιμής της συνάρτησης αξιολόγησης με τιμή Crossover Rate ίση με 0.8

Πηγή: Matlab

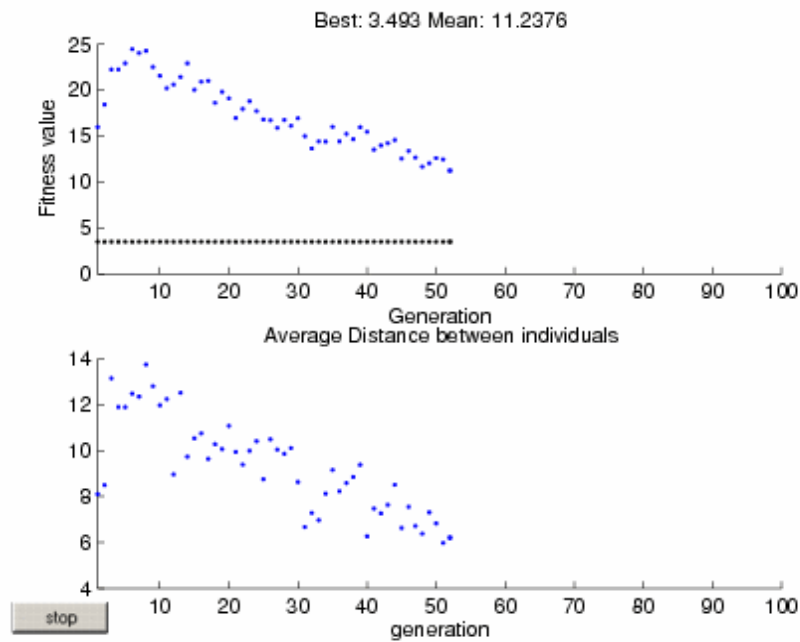
Παρατηρείται ότι η ελάχιστη τιμή της συνάρτησης αξιολόγησης είναι 0.23492.

Αν εφαρμοστεί όμως ο ΓΑ με τιμή CrossoverFraction ίση με 1 που σημαίνει ότι δεν υφίσταται καθόλου μετάλλαξη είτε με τιμή 0 που σημαίνει ότι δεν υφίσταται καθόλου διασταύρωση τότε προκύπτουν:



Εικόνα 2.7 - Διάγραμμα καλύτερης και μέσης τιμής της συνάρτησης αξιολόγησης με τιμή Crossover Rate ίση με 1

Πηγή: Matlab



Εικόνα 2.8 - Διάγραμμα καλύτερης και μέσης τιμής της συνάρτησης αξιολόγησης με τιμή Crossover Rate ίση με 0
Πηγή: Matlab

Από τα διαγράμματα των εικόνων 2.6, 2.7 και 2.8 προκύπτει η μη σωστή λειτουργία του ΓΑ στις περιπτώσεις αυτές καθώς η ελάχιστη (καλύτερη) τιμή της συνάρτησης αξιολόγησης είναι ίση με 1,3161 στην πρώτη περίπτωση και ίση με 3,493 στην δεύτερη, εντελώς διαφορετικές από την τιμή που προέκυψε από την αρχική εκτέλεση του ΓΑ.

Η σύνταξη που πρέπει να έχει η συνάρτηση για καθορισμό ρυθμού διασταύρωσης για παράδειγμα 0,8 πρέπει να είναι η ακόλουθη:

```
ga.setCrossoverRate(0.85)
```

2.3.10 Ανάλυση της Επιλογής Mutator Methods

Η επιλογή αυτή καθορίζει τον τρόπο μετάλλαξης των ατόμων μιας γενιάς, δηλαδή τον τρόπο που ο ΓΑ πραγματοποιεί μικρές τυχαίες αλλαγές σε κάποια από τα άτομα του πληθυσμού, ώστε να δημιουργηθούν παιδιά με μετάλλαξη. Η μετάλλαξη παρέχει γενετική ποικιλότητα ενώ καθιστά δυνατή την αναζήτηση σε ένα ευρύτερο χώρο. Διαθέσιμες συναρτήσεις για την εφαρμογή της επιλογής αυτής είναι:

- **Allele** ποικιλομορφία: Η μέθοδος αυτή εφαρμόζεται στην περίπτωση που χρησιμοποιείται διαφορετικό εύρος τιμών στις παραμέτρους του προβλήματος. Η μετάλλαξη υλοποιείται ανάμεσα στις διαθέσιμες τιμές της κάθε παραμέτρου, ανάλογα με τον ρυθμό μετάλλαξης που αναλύεται παρακάτω.
- **IntegerGaussian**: Η λειτουργία της πραγματοποιείται με την πρόσθεση ενός τυχαίου ακέραιου αριθμού που λαμβάνεται από την γκαουσιανή κατανομή με μέση τιμή το 0, για κάθε είσοδο του διανύσματος γονέα. Επιπλέον μπορεί να δεχτεί ως παράμετρο το μέσο όρο και την τυπική απόκλιση της τυχαίας κατανομής.
- **RealGaussian**: Η λειτουργία της πραγματοποιείται με την πρόσθεση ενός τυχαίου πραγματικού αριθμού που λαμβάνεται από την γκαουσιανή κατανομή με μέση τιμή το

0, για κάθε είσοδο του διανύσματος γονέα. Επιπλέον μπορεί να δεχτεί ως παράμετρο το μέσο όρο και την τυπική απόκλιση της τυχαίας κατανομής.

- **IntegerRange:** Η μέθοδος αυτή χρησιμοποιεί το ακέραιο μέρος του κάθε γονιδίου των γονέων, υλοποιεί μετάλλαξη ανάλογα με την δοσμένη πιθανότητα και δημιουργεί τους απογόνους.
- **RealRange:** Η μέθοδος αυτή χρησιμοποιεί το ακέραιο μέρος του κάθε γονιδίου των γονέων, υλοποιεί μετάλλαξη ανάλογα με την δοσμένη πιθανότητα και δημιουργεί τους απογόνους
- **SIM:** Αυτή η λειτουργία λαμβάνει τα άτομα του πληθυσμού, μεταλλάσσει κάθε άτομο αντιστρέφοντας τις θέσεις των μεταβλητών ανάλογα με την δεδομένη πιθανότητα και επιστρέφει το αποτέλεσμα του απογόνου που προκύπτει.
- **Swap:** Η μέθοδος αυτή μεταλλάσσει κάθε άτομο του τρέχοντος πληθυσμού με εναλλαγή των μεταβλητών του ατόμου, ανάλογα με την δεδομένη πιθανότητα και επιστρέφει το αποτέλεσμα του απογόνου που προκύπτει.

Η σύνταξη που πρέπει να έχει η συνάρτηση για παράδειγμα στην επιλογή **Swap** πρέπει να είναι η ακόλουθη:

```
genome.mutator.set(Mutators.G1DListMutatorSwap)
```

2.3.11 Ανάλυση της Επιλογής Mutation Rate

Η επιλογή αυτή, καθορίζει το ποσοστό του εκάστοτε πληθυσμού, εκτός από τα παιδιά που περιλαμβάνονται στην επιλογή **Elite**, το οποίο ουσιαστικά προέρχονται από μετάλλαξη. Η προκαθορισμένη τιμή για αυτήν την επιλογή είναι 0.02 δηλαδή 2%. Η καλύτερη επιλογή κυμαίνεται από 0,5% - 3%. Μια πολύ μικρή τιμή στο ποσοστό μετάλλαξης μπορεί να οδηγήσει σε γενετική παρέκκλιση. Αντίθετα μια μεγάλη τιμή μπορεί να οδηγήσει σε απώλεια σωστής λύσης εκτός και αν υφίσταται η ελιτιστική επιλογή.

Η σύνταξη που πρέπει να έχει η συνάρτηση για καθορισμό ρυθμού μετάλλαξης για παράδειγμα 0,025 πρέπει να είναι η ακόλουθη:

```
ga.setMutationRate(0.025)
```

2.3.12 Ανάλυση των Βασικών Επιλογών Migration

Οι επιλογές **Migration** (μετανάστευσης) διευκρινίζουν πώς τα άτομα κινούνται μεταξύ των υποπληθυσμών. Η μετανάστευση εμφανίζεται εάν το μέγεθος πληθυσμού είναι μεγαλύτερο από 1. Όταν η μετανάστευση εμφανίζεται, τα καλύτερα άτομα από ένα υποσύνολο πληθυσμού αντικαθιστούν τα χειρότερα άτομα σε ένα άλλο υποσύνολο πληθυσμού. Τα άτομα που μεταναστεύουν από ένα υποσύνολο πληθυσμού σε ένα άλλο αντιγράφονται και δεν αφαιρούνται από τον υποπληθυσμό πηγή.

2.3.12.1 MigrationSelector

Μέσω της επιλογής αυτής καθορίζεται η μέθοδος επιλογής των ατόμων που θα μεταναστεύσουν. Οι μέθοδοι είναι αυτές που αναλύθηκαν στο εδάφιο 4.5. Η αλλαγή της μεθόδου για παράδειγμα της **GRouletteWheel** υλοποιείται με την εφαρμογή της εντολής:

```
migration_scheme.selector.set(Selectors.GRouletteWheel)
```

2.3.12.2 MigrationRate

Μέσω της επιλογής αυτής καθορίζεται ανά πόσες γενιές θα πραγματοποιείται μετανάστευση. Για παράδειγμα εάν τεθεί τιμή 20 αυτό σημαίνει ότι ανά 20 γενιές λαμβάνει χώρα μετανάστευση. η προκαθορισμένη τιμή είναι 20. Η επιλογή αυτή π.χ. για τιμή ίση με 25, γίνεται με την εφαρμογή της εντολής:

```
ga.setMigrationRate(25)
```

2.3.12.3 NumIndividuals

Η επιλογή αυτή καθορίζει πόσα άτομα θα μετακινούνται μεταξύ των υποσυνόλων πληθυσμού. Η τιμή της επιλογής αυτής διευκρινίζει το πλήθος των ατόμων του μικρότερου των δύο υποσυνόλων πληθυσμού που μεταναστεύουν. Η επιλογή αυτή υλοποιείται με την εφαρμογή της εντολής:

```
ga.setNumIndividuals(num_individuals)
```

2.3.13 Ανάλυση των Επιλογών - Συνθηκών Τερματισμού του ΓΑ

Στο συγκεκριμένο εδάφιο αναλύονται οι επιλογές εκείνες, των οποίων τις τιμές χρησιμοποιεί ο ΓΑ, ώστε να τερματιστεί σε περίπτωση που υλοποιηθεί κάποια από αυτές.

2.3.13.1 Ανάλυση της επιλογής Generations

Μέσω της επιλογής αυτής καθορίζεται το πλήθος των γενεών το οποίο θα δημιουργήσει ο ΓΑ και για καθεμιά από τις οποίες θα παράγεται νέος πληθυσμός. Εάν π.χ. οριστεί το πλήθος των γενεών ως 100 τότε ο ΓΑ θα τρέξει για 100 γενιές και εάν δεν τερματιστεί λόγω κάποιου από τα παρακάτω κριτήρια θα σταματήσει εξαιτίας της εκτέλεσης του για το μέγιστο αριθμό γενεών.

Η προκαθορισμένη τιμή είναι 100 γενεές ενώ η εντολή που δίνεται ώστε να προσδιοριστεί π.χ. πλήθος γενεών 15 είναι:

```
ga.setGenerations(15)
```

2.3.13.2 Ανάλυση της επιλογής ConvergeCriteria

Η επιλογή αυτή σταματά την εξέλιξη όταν ο πληθυσμός θεωρείται ότι συγκλίνει. Ως σύγκλιση πληθυσμού θεωρείται όταν ο μέσος όρος της συνάρτησης αξιολόγησης του συγκεκριμένου πληθυσμού γίνει μικρότερος από την προκαθορισμένη τιμή .

Η συγκεκριμένη επιλογή υλοποιείται μέσω της εντολής

```
ga.terminationCriteria.set(GSimpleGA.ConvergenceCriteria)
```

2.3.13.3 Ανάλυση της επιλογής FitnessStatsCriteria

Η επιλογή αυτή σταματά την εξέλιξη όταν ο υπάρξει σύγκλιση στην τιμή της συνάρτησης αξιολόγησης. Ουσιαστικά η σύγκλιση υλοποιείται με βάση τα στατιστικά της συνάρτησης αξιολόγησης, όταν δηλαδή επιτευχθεί η προκαθορισμένη τιμή γι αυτά.

Η συγκεκριμένη επιλογή υλοποιείται μέσω της εντολής

```
ga.terminationCriteria.set(GSimpleGA.FitnessStatsCriteria)
```

2.3.14 Ανάλυση της Επιλογής Fitness Scaling Methods

Η επιλογή αυτή έχει ως στόχο να μετατρέψει τις αρχικές τιμές της συνάρτησης αξιολόγησης εντός ενός εύρους τιμών το οποίο να είναι κατάλληλο για την σωστή λειτουργία της συνάρτησης επιλογής. Το εύρος της κλίμακας τιμών μετά την εφαρμογή της επιλογής αυτής επηρεάζει την απόδοση του ΓΑ. Αν η κλίμακα τιμών έχει μεγάλο εύρος, τα άτομα με τις υψηλότερες τιμές αναπαράγονται πολύ γρήγορα, η δεξαμενή γονιδίων είναι συγκεκριμένη και αποτρέπεται η αναζήτηση του ΓΑ σε άλλες περιοχές, με κίνδυνο αποκλεισμού σε κάποιο τοπικό ελάχιστο. Από την άλλη πλευρά, αν η κλίμακα τιμών ποικίλει ελάχιστα, τότε όλα τα άτομα έχουν περίπου την ίδια ευκαιρία αναπαραγωγής και η αναζήτηση θα προχωρήσει με πολύ αργούς ρυθμούς.

Οι διαθέσιμες συναρτήσεις για την εφαρμογή της επιλογής αυτής είναι: *Linear*, *PowerLaw*, *Exponential*, *Saturated SigmaTrunc* και *Boltzmann*.

2.3.15 Ανάλυση της Επιλογής DBAdapters

Μέσω της επιλογής αυτής δημιουργείται κατά την διαδικασία εκτέλεσης του ΓΑ μια βάση στην οποία περιέχονται όλα τα στατιστικά αποτελέσματα των γενεών. Εν συνεχεία η βάση αυτή μπορεί να χρησιμοποιηθεί προκειμένου να εκτυπωθούν στατιστικά διαγράμματα.

Η εντολή που δίνεται για την δημιουργία της βάσης είναι η εξής:

```
sqlite_adapter = DBAdapters.DBSQLite(identify="ex1")
ga.setDBAdapter(sqlite_adapter)
```

2.3.16 Ανάλυση της Επιλογής Graphical Plotting Tool Options

Μέσω της επιλογής αυτής δίνεται η δυνατότητα στον χρήστη να μπορεί να δημιουργεί διαγράμματα στα οποία να εμφανίζονται στατιστικά από την διαδικασία εκτέλεσης του ΓΑ. Απαραίτητη προϋπόθεση είναι να έχει δημιουργηθεί η βάση σύμφωνα με το εδάφιο 2.3.15. Εν συνεχεία σε περιβάλλον DOS εκτελώντας την εντολή που εμφανίζεται παρακάτω και δίνοντας τον κατάλληλο αριθμό (πχ -1 για raw scores, -5 για min/max raw και fitness scores, λαμβάνει το εκάστοτε διάγραμμα.

Η εντολή που δίνεται για εκτύπωση π.χ. δυο συναρτήσεων από τις παραπάνω είναι η εξής:

```
pyevolve_graph.py -i ex1 -1
```

2.4 Ανάλυση των Αλγορίθμων Κατάτμησης

2.4.1 Γενικά

Όπως έχει ήδη αναφερθεί οι μέθοδοι κατάτμησης μιας εικόνας διακρίνονται σε αυτές που βασίζονται στην εύρεση ορίων και σε αυτές που βασίζονται στην εύρεση περιοχών. Στην παρούσα διπλωματική χρησιμοποιήθηκαν αλγόριθμοι κατάτμησης που βασίζονται στην εύρεση περιοχών, υλοποιημένοι στην γλώσσα python. Συγκεκριμένα χρησιμοποιήθηκε αλγόριθμος που βασίζεται στον μετασχηματισμό του υδροκρίτη (watershed), αλγόριθμος που βασίζεται σε τεχνικές τοπικής ανάλυσης επέκτασης περιοχών (region growing) και σε αλγόριθμο που βασίζεται σε τεχνικές τοπική ένωσης περιοχών (region merging).

2.4.2 Αλγόριθμος βασισμένος στον μετασχηματισμό του Υδροκρίτη (Watershed)

Ο αλγόριθμος αυτός χρησιμοποιεί έναν μορφολογικό μετασχηματισμό για τον προσδιορισμό περιοχών στην εικόνα. Οι γραμμές των ορίων του υδροκρίτη χωρίζουν μεμονωμένες λεκάνες απόθεσης. Εάν υπολογιστεί η κλίση με βάση τους τόνους του γκρι μιας εικόνας, παράγεται μια εικόνα κλίσεων. Τότε η εικόνα μπορεί να ερμηνευτεί σαν μια τοπογραφική τρισδιάστατη επιφάνεια, στην οποία οι τιμές της κλίσης του γκρι, αντιστοιχούν στις τιμές του υψόμετρου. Επομένως, οι ακμές των περιοχών αντιστοιχούν σε ψηλούς υδροκρίτες και οι περιοχές χαμηλών κλίσεων αντιστοιχούν σε λεκάνες απόθεσης.

Στόχος της κατάτμησης που βασίζεται σε περιοχές είναι να δημιουργήσει ομοιογενείς περιοχές. Στο μετασχηματισμό του υδροκρίτη, οι λεκάνες απόθεσης της τοπογραφικής επιφάνειας είναι ομοιογενείς με την έννοια ότι όλα τα εικονοστοιχεία που ανήκουν στην ίδια λεκάνη απόθεσης, συνδέονται με το ελάχιστο υψόμετρο της περιοχής με μια σειρά εικονοστοιχείων που παρουσιάζουν σταδιακή μείωση υψόμετρου κατά μήκος της διαδρομής. Οι λεκάνες αυτού του είδους, αντιπροσωπεύουν τις περιοχές της κατάτμησης της εικόνας.

Ο αλγόριθμος υλοποίησης χωρίζεται σε πέντε τμήματα τα οποία και διαχωρίζονται μεταξύ τους. Στον συγκεκριμένο αλγόριθμο χρησιμοποιούνται τέσσερις παράμετροι τις οποίες προσπαθεί να βελτιστοποιήσει ο ΓΑ. Αυτές αναφέρονται ως `genome[0]` - `genome[3]`.

```

A ΤΜΗΜΑ
dataset=gdal.Open('Mars_Green.img', GA_ReadOnly)
dna=dataset.ReadAsArray(0,0,dataset.RasterXSize,
dataset.RasterYSize)

B ΤΜΗΜΑ
T = pymorph.threshad(dna,genome[0],genome[1])
T = T.astype(numpy.uint8)

Γ ΤΜΗΜΑ
dnaf = ndimage.gaussian_filter(dna,genome[2])
rmax = pymorph.regmax(dnaf)
seeds,nr_nuclei = ndimage.label(rmax)

Δ ΤΜΗΜΑ
se=numpy.ones((genome[3],genome[3]))
#se = ([1,1,1],[1,1,1],[1,1,1])
gr = ndimage.morphology.morphological_gradient(T,footprint=se)

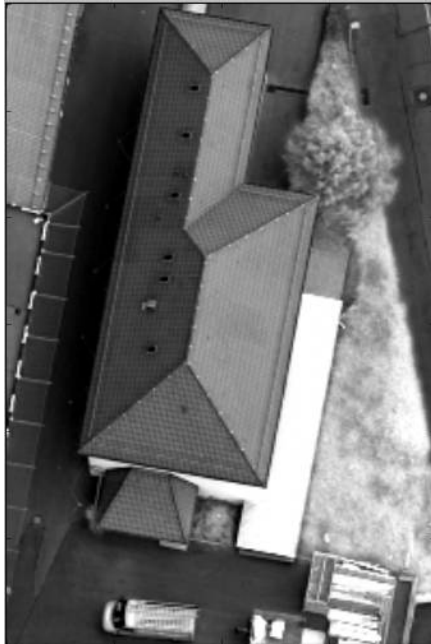
Ε ΤΜΗΜΑ
im = Image.fromarray(pymorph.cwatershed(gr, seeds))
im.save('segment_image.tif')

```

Στο πρώτο τμήμα ο αλγόριθμος δέχεται ως είσοδο ένα κανάλι της εικόνας ή μια παγχρωματική εικόνα. Υλοποιεί μετατροπή σε πίνακα, με στοιχεία τις τιμές των εικονοστοιχείων της εικόνας.

Στο δεύτερο τμήμα εφαρμόζεται κατωφλίωση στον πίνακα όπου τιμή 1 λαμβάνουν τα στοιχεία ανάμεσα στις τιμές `genome[0]` και `genome[1]` και τιμή 0 τα υπόλοιπα. Με την εφαρμογή αυτή δημιουργείται ο πίνακας «T».

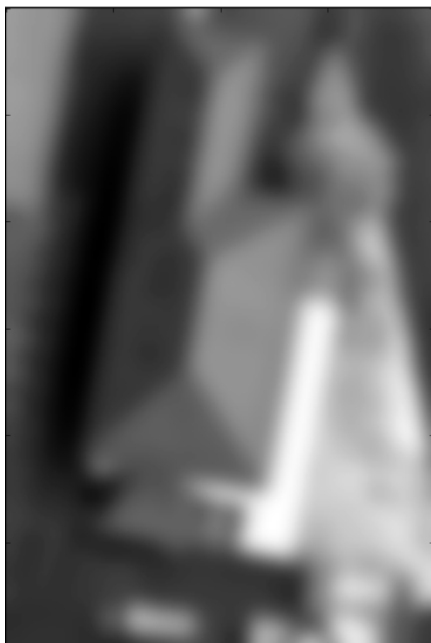
Στο τρίτο τμήμα υλοποιείται η εφαρμογή φίλτρου gauss στην αρχική εικόνα, το οποίο χρησιμοποιείται για την εξομάλυνση της. Η διάσταση του δομικού στοιχείου που χρησιμοποιείται αποτελεί την τρίτη παράμετρο του ΓΑ (`genome[2]`). Στις εικόνες 2.10, 2.11 και 2.12 εμφανίζεται η εφαρμογή του φίλτρου με χρήση πίνακα διαστάσεων 3 x 3, 5 x 5 και 7 x 7 αντίστοιχα. Στο αποτέλεσμα αναζητούνται τοπικά μέγιστα τα οποία όσα ακολουθούν την μορφή του εκάστοτε δομικού στοιχείου της συνάρτησης *label*, λαμβάνουν μια τιμή κατά αύξουσα σειρά, δημιουργώντας ουσιαστικά αντικείμενα στην εικόνα.



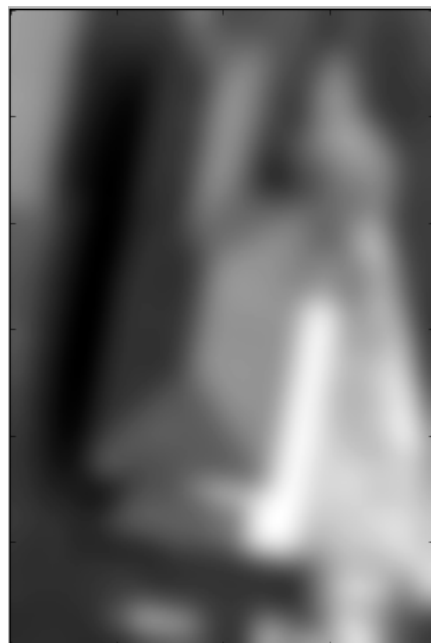
Εικόνα 2.9 - Αρχική εικόνα



Εικόνα 2.10 - Εφαρμογή φίλτρου Gauss 3 x 3



Εικόνα 2.11 - Εφαρμογή φίλτρου Gauss 5 x 5



Εικόνα 2.12 - Εφαρμογή φίλτρου Gauss 7 x 7

Στο τέταρτο τμήμα υλοποιείται μορφολογική παράγωγος δηλαδή διαφορά ανάμεσα στην εφαρμογή διαστολής και συστολής στο πίνακα «T» που προέκυψε στο δεύτερο τμήμα, με δομικό στοιχείο ένα πίνακα με άσσους σε όλες τις θέσεις του, η διάσταση του οποίου αποτελεί και την τέταρτη παράμετρο του ΓΑ (genome[3]).

Τέλος στο πέμπτο τμήμα υλοποιείται ο αλγόριθμος watershed με παραμέτρους τα αποτελέσματα των τμημάτων τρία και τέσσερα και αποθηκεύεται το αποτέλεσμα της κατάτμησης.

2.4.3 Αλγόριθμος βασισμένος σε τεχνικές τοπικής ανάλυσης επέκτασης περιοχών (region growing)

Η μέθοδος της κατωφλίωσης, ενώ έχει σαν αποτέλεσμα την κατάτμηση της εικόνας σε περιοχές, δεν λαμβάνει υπόψη καθόλου τις χωρικές σχέσεις των εικονοστοιχείων κατά την διαδικασία της επιλογής. Ο αλγόριθμος αύξησης περιοχών αντίθετα, για να συμπεριλάβει ένα αντικείμενο σε μια περιοχή, ελέγχει αρχικά τις χωρικές σχέσεις των εικονοστοιχείων της εικόνας και στη συνέχεια κάποιο κριτήριο ένωσης των εικονοστοιχείων όπως για παράδειγμα την μικρή διαφορά φασματικών τιμών.

Η λογική του αλγορίθμου αυτού βασίζεται στην έναρξη της διαδικασίας από ένα συγκεκριμένο σημείο πάνω στην εικόνα ή τυχαίες θέσεις πάνω σε αυτή. Τα συγκεκριμένα αυτά σημεία μπορεί να προέρχονται είτε από άλλα στάδια επεξεργασίας, είτε από στατιστική ανάλυση των εικόνων, είτε από κάποιον μετασχηματισμό.

Μετά την επιλογή του ή των αρχικών σημείων, ο αλγόριθμος προχωράει στην σειριακή επεξεργασία του κάθε σημείου σπόρου (seed point) στην αρχική εικόνα. Ορίζεται ένα κριτήριο συνένωσης όπως οι διαφορές τόνων του γκρι, το οποίο καθορίζει εφόσον μπορεί να πραγματοποιηθεί μια συνένωση ή όχι. Ακολουθεί μια επαναληπτική διαδικασία κατά την οποία ελέγχονται τα εικονοστοιχεία που γειτνιάζουν με τα αρχικά σημεία και με βάση το κριτήριο ένωσης, αυξάνεται το εμβαδόν του αντικειμένου, συγχωνεύοντας νέα εικονοστοιχεία. Συνολικά, η όλη διαδικασία επαναλαμβάνεται μέχρι που να μην μπορούν να πραγματοποιηθούν πλέον συγχωνεύσεις σε όλη την επιφάνεια της εικόνας. Σε αυτό το στάδιο θα έχουν προκύψει το πολύ ισάριθμα αντικείμενα με τα αρχικά σημεία, ενώ τα εικονοστοιχεία που δεν συγχωνεύτηκαν, αποτελούν το υπόβαθρο της κατάτμησης. Αυτό σημαίνει ότι με την αύξηση περιοχών δεν επιτυγχάνεται πλήρης κατάτμηση της εικόνας σε αντικείμενα.

Ο αλγόριθμος υλοποίησης παρουσιάζεται στο πλαίσιο που ακολουθεί. Στον συγκεκριμένο αλγόριθμο χρησιμοποιούνται τρεις παράμετροι τις οποίες προσπαθεί να βελτιστοποιήσει ο ΓΑ. Αυτές αναφέρονται ως genome[0] - genome[2]. Η πρώτη αφορά την τιμή κατωφλίωσης ενώ οι δυο επόμενες την συντεταγμένη x και y του εικονοστοιχείου της εικόνας από το οποίο ξεκινάει η διαδικασία.

```

img_name = "toposys.tif"
threshold = genome[0]
connectivity = 8
img = cv.LoadImage(img_name, cv.CV_LOAD_IMAGE_GRAYSCALE)
seed = (genome[1], genome[2])
out_img = se.simple_region_growing(img, seed, threshold,
connectivity)
cv.SaveImage("segment_image.tif ",out_img)

segment_tif = "segment_image.tif"
segment_vector_shp = "segment_2_vector.shp"
segment_final_tif = "segment_final.tif"

arcpy.env.overwriteOutput = True

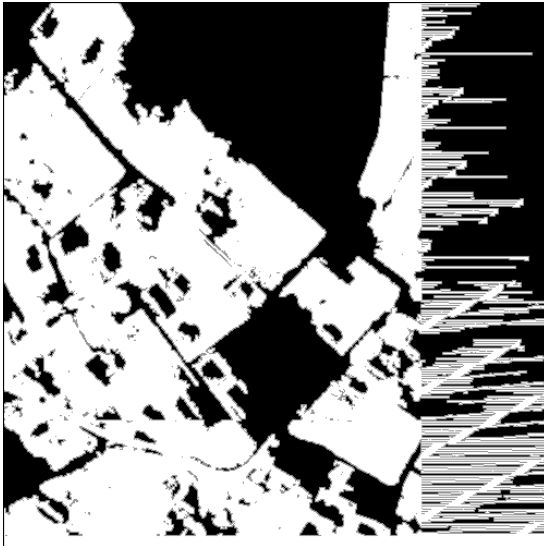
# Process: Raster to Polygon
arcpy.RasterToPolygon_conversion(segment_tif, segment_vector_shp,
"SIMPLIFY", "Value")

# Process: Polygon to Raster
arcpy.PolygonToRaster_conversion(segment_vector_shp,"ID",
segment_final_tif, "CELL_CENTER", "NONE", segment_tif)

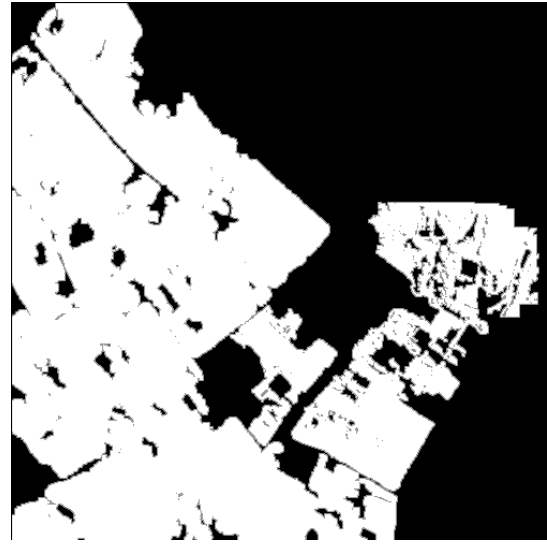
```

Στον αλγόριθμο αρχικά δίνεται η εικόνα στην οποία θα υλοποιηθεί η κατάτμηση. Εν συνεχεία επιλέγεται με βάση τον γενετικό αλγόριθμο η τιμή της κατωφλίωσης. Ακολουθεί η παράμετρος η οποία ορίζει αν στην διαδικασία της κατάτμησης θα χρησιμοποιηθούν οι τιμές από τα οκτώ γειτονικά ή τα τέσσερα γειτονικά εικονοστοιχεία. Εν συνεχεία η εικόνα μετατρέπεται σε εικόνα τόνων του γκρι. Επιλέγεται το σημείο έναρξης της διαδικασίας με βάση τις δυο επόμενες τιμές του αλγορίθμου κατάτμησης. Εν συνεχεία υλοποιείται ο αλγόριθμος κατάτμησης και αποθηκεύεται το αποτέλεσμα της κατάτμησης. Το αποτέλεσμα της κατάτμησης ("segment.tif") δεν χρησιμοποιείται αυτούσιο για την διεργασία του ελέγχου της ακρίβειας μέσω της συνάρτησης αξιολόγησης, καθώς η εικόνα κατάτμησης που προκύπτει δεν είναι στην μορφή που απαιτείται για να συνεχιστεί η διαδικασία της αξιολόγησης δηλαδή το κάθε πολύγωνο να έχει μια μοναδική χαρακτηριστική αύξουσα αρίθμηση. Για το λόγο αυτό η εικόνα κατάτμησης ("segment.tif") μετατρέπεται σε διανυσματικό αρχείο με πολύγωνα τα αντικείμενα της κατάτμησης ("segment_2_vector.shp") και εν συνεχεία το αποτέλεσμα αυτό μετατρέπεται με βάση το ID σε εικόνα ("segment_final.tif") η οποία έχει την απαιτούμενη μορφή ώστε να συνεχιστεί η διαδικασία της κατάτμησης. Τα παραπάνω υλοποιούνται με χρήση της βιβλιοθήκης arcpy του ArcGIS.

Επιπλέον παρατηρείται ότι η χρήση εικόνων τύπου .tif, με τον συγκεκριμένο αλγόριθμο εμφανίζει προβλήματα στο αποτέλεσμα της κατάτμησης καθώς σε ένα τμήμα της εικόνας δεν δημιουργούνται ορθά αντικείμενα, όπως παρουσιάζεται και στην εικόνα 2.13. Αντίθετα σε περίπτωση χρησιμοποίησης εικόνων τύπου .png αντι .tif, εξαλείφεται το παραπάνω πρόβλημα, όπως παρουσιάζεται στην εικόνα 2.14.



Εικόνα 2.13 - Αποτέλεσμα κατάτμησης με
χρησιμοποίηση εικόνας μορφής .tif



Εικόνα 2.14 - Αποτέλεσμα κατάτμησης με
χρησιμοποίηση εικόνας μορφής .png

Μια επιπλέον παρατήρηση για τον συγκεκριμένο αλγόριθμο είναι ότι σε ορισμένες τιμές των παραμέτρων, η εφαρμογή του οδηγούσε στο σφάλμα που εμφανίζεται παρακάτω.

```
Traceback (most recent call last):
```

```
File
```

```
"D:\EMP\geoplirforiki\diplomatiki\code_genetic_algorithms\now\region_growing.py",  
line 11, in <module>
```

```
    out_img = se.simple_region_growing(img, seed, threshold, connectivity)
```

```
File
```

```
"D:\EMP\geoplirforiki\diplomatiki\code_genetic_algorithms\now\tippy\segmentations.  
py", line 88, in simple_region_growing
```

```
    dist = abs(int(numpy.mean(contour_val)) - mean_reg)
```

```
ValueError: cannot convert float NaN to integer
```

Η επίλυση του συγκεκριμένου σφάλματος υλοποιήθηκε με αλλαγή στον κώδικα της συνάρτησης του αρχείου `segmentations.py` που χρησιμοποιούσε ο συγκεκριμένος αλγόριθμος.

2.4.4 Αλγόριθμος που βασίζεται σε τεχνικές τοπική ένωσης περιοχών (region merging).

Οι αλγόριθμοι κατάτμησης εικόνας με συνένωση περιοχών είναι από τους πιο διαδεδομένους στην βιβλιογραφία και χρησιμοποιούνται εκτεταμένα στην Τηλεπισκόπηση και ειδικότερα στην αντικειμενοστραφή ανάλυση Εικόνας. Σε αυτή την οικογένεια αλγορίθμων ουσιαστικά ορίζεται ένα χαρακτηριστικό μέγεθος που εκφράζει την ομοιογένεια περιοχών. Ο βασικός στόχος αυτών των αλγορίθμων είναι ο διαχωρισμός της εικόνας σε αντικείμενα για τα οποία όμως η ομοιογένεια να είναι μέγιστη. Το κριτήριο της ομοιογένειας μπορεί να βασίζεται στο επίπεδο του τόνου του γκριζού, στο χρώμα, στην υφή, στο σχήμα ή ακόμα και σε κάποιο μοντέλο σημασιολογικής πληροφορίας.

Οι περιοχές που θα προκύψουν από την κατάτμηση της εικόνας θα πρέπει να είναι ταυτόχρονα ομοιογενής και μέγιστες, και με τον όρο μέγιστες εννοείται ότι το κριτήριο ομοιογένειας θα πρέπει να μην είναι αληθές, για κανένα γειτονικό αντικείμενο.

Η πιο φυσική από τις μεθόδους κατάτμησης εικόνας με βάση περιοχές είναι η διαδικασία κατά την οποία η εικόνα θεωρείται σαν ένα σύνολο από περιοχές, οι οποίες αρχικά αποτελούνται από μεμονωμένα εικονοστοιχεία. Οι αρχικές περιοχές αρχίζουν να ενώνονται με τον συγκεκριμένο κριτήριο έως ότου σταματήσει η διαδικασία της συνένωσης με βάση τα κριτήρια που τίθενται κάθε φορά.

Ο αλγόριθμος υλοποίησης παρουσιάζεται στο πλαίσιο που ακολουθεί. Στον συγκεκριμένο αλγόριθμο χρησιμοποιούνται δυο παράμετροι τις οποίες προσπαθεί να βελτιστοποιήσει ο ΓΑ. Αυτές αναφέρονται ως `genome[0]` και `genome[1]`. Ουσιαστικά το άθροισμα των δυο παραμέτρων αποτελεί την παράμετρο ελέγχου της κλίμακας της κατάτμησης. Ωστόσο για λόγους καλής λειτουργίας του ΓΑ η παράμετρος αυτή αποτελείται από το άθροισμα των δυο παραμέτρων του ΓΑ.

```

im = imread("toposys.png")
print genome[0]
print genome[1]
srm = SRM(im, genome[0]+genome[1])
segmented = srm.run()
imsave('segment.tif', segmented)

arcpy.env.workspace = "
D:\EMP\geoplirforiki\diplomatiki\code_genetic_algorithms\region_mer
ging"

# Local variables:
Band_1 = "segment.tif\Band_1"
segment_band1_tif = "segment_band1.tif"
segment_tif = "segment_band1.tif"
segment_vector_shp = "segment_2_vector.shp"
segment_final_tif = "segment_final.tif"

arcpy.env.overwriteOutput = True

# Process: Composite Bands
arcpy.CompositeBands_management(Band_1, segment_band1_tif)

# Process: Raster to Polygon
arcpy.RasterToPolygon_conversion(segment_tif, segment_vector_shp,
"SIMPLIFY", "Value")

# Process: Polygon to Raster
arcpy.PolygonToRaster_conversion(segment_vector_shp, "ID",
segment_final_tif, "CELL_CENTER", "NONE", segment_tif)

```

Στον κώδικα παραπάνω το αποτέλεσμα της κατάτμησης ("segment.tif") δεν χρησιμοποιείται αυτούσιο για την διεργασία του ελέγχου της ακρίβειας μέσω της συνάρτησης αξιολόγησης, καθώς η εικόνα κατάτμησης που προκύπτει είναι έγχρωμη (Κόκκινο, Μπλε και Πράσινο κανάλι) και δεν είναι στην μορφή που απαιτείται για να συνεχιστεί η διαδικασία της αξιολόγησης δηλαδή το κάθε πολύγωνο να έχει μια μοναδική

χαρακτηριστική αύξουσα αρίθμηση. Για το λόγο αυτό το ένα κανάλι της προκύπτουσας εικόνας ("segment_band1.tif") μετατρέπεται σε διανυσματικό αρχείο με πολύγωνα τα αντικείμενα της κατάτμησης ("segment_2_vector.shp") και εν συνεχεία το αποτέλεσμα αυτό μετατρέπεται με βάση το ID σε εικόνα ("segment_final.tif") η οποία έχει την απαιτούμενη μορφή ώστε να συνεχιστεί η διαδικασία της κατάτμησης. Τα παραπάνω υλοποιούνται με χρήση της βιβλιοθήκης arcpy του ArcGIS.

2.5 Προηγούμενες Μελέτες

Κατά καιρούς έχουν πραγματοποιηθεί διάφορες προσπάθειες εφαρμογής γενετικού προγραμματισμού (ΓΠ) στο πεδίο της τηλεπισκόπησης και γενικότερα σε διάφορων ειδών εικόνες, ώστε να αυτοματοποιηθεί όσο το δυνατόν περισσότερο η επεξεργασία εικόνων και η αυτόματη εξαγωγή χαρακτηριστικών από αυτές. Μερικές από αυτές παρουσιάζονται παρακάτω.

Ο **Tackett** (1993), χρησιμοποίησε ΓΠ για την ανάπτυξη ενός δέντρου επεξεργασίας για την ταξινόμηση των χαρακτηριστικών που εξάγονται από εικόνες. Συγκεκριμένα στόχος της εργασίας αποτέλεσε η ταξινόμηση χαρακτηριστικών διανυσμάτων που προέρχονταν από IR εικόνες σε στόχους ή μη, χρησιμοποιώντας μια βάση δεδομένων με 2000 διαφορετικά δεδομένα εκπαίδευσης. Η απόδοση του αλγορίθμου ελέγχθηκε σε μια βάση 7000 δειγμάτων. Η εργασία αυτή αποτέλεσε καινοτομία σε σύγκριση με τα προβλήματα για τα οποία είχε εφαρμοστεί ο ΓΠ μέχρι τότε. Συγκεκριμένα εκτελέστηκαν δυο πειράματα. Στην πρώτη σειρά πειραμάτων χρησιμοποιήθηκαν κλασικά χαρακτηριστικά εικόνων και ελαχιστοποιήθηκε η λανθασμένη κατάταξη τόσο των στόχων όσο και των μη στόχων. Στη δεύτερη σειρά πειραμάτων, μέσω του ΓΠ επιτράπηκε η δημιουργία ομάδας χαρακτηριστικών από πρωτογενείς μετρήσεις της έντασης του μέσου όρου και της τυπικής απόκλισης για τις εικόνες. Για λόγους σύγκρισης, τα ίδια δεδομένα εκπαίδευσης και δοκιμής χρησιμοποιήθηκαν για την εκπαίδευση δυο άλλων συστημάτων ταξινόμησης, το δυαδικό δέντρο ταξινομητή και το νευρωνικό δίκτυο οπιθοδιάδοσης. Το δίκτυο του ΓΠ επιτύγχανε υψηλότερες επιδόσεις με μειωμένες υπολογιστικές απαιτήσεις.

Οι **Harris και Buxton** (1996), χρησιμοποίησαν ΓΠ για την δημιουργία ανιχνευτών ακμών σε μονοδιάστατα σήματα και σε χαρακτηριστικά εικόνων. Ουσιαστικά μέσω την ανίχνευσης ακμών αναζητούνταν τυχόν ασυνέχειες σε σήματα και εικόνες. Η μέθοδος αυτή, για την οποία υπήρχε η πρόθεση να επεκταθεί στην ανάπτυξη πρακτικών ανιχνευτών ακμών, για χρήση σε επεξεργασία εικόνας και όρασης υπολογιστών, χρησιμοποιεί θεωρητικά μέτρα απόδοσης ως κριτήρια για το σχεδιασμό του πειράματος.

Ο **Poli** (1996), χρησιμοποίησε ΓΠ με στόχο να δημιουργήσει αποτελεσματικά φίλτρα εικόνων. Τα φίλτρα αυτά μπορούσαν να χρησιμοποιηθούν είτε για την ενίσχυση και στην συνέχεια ανίχνευση χαρακτηριστικών ενδιαφέροντος σε εικόνες είτε για την κατασκευή αλγορίθμων κατάτμησης βασισμένους σε ταξινόμηση των pixel της εικόνας. Έγιναν πειράματα με ιατρικές εικόνες που απέδειξαν την αποτελεσματικότητα της προσέγγισης αυτής.

Οι **Stanhope και Daida** (1998), χρησιμοποίησαν τον ΓΠ σε δυο εργασίες. Η πρώτη είχε σκοπό την δημιουργία κανόνων για ταξινόμηση στόχων σε ένα σετ από εικόνες SAR και η δεύτερη την δημιουργία κανόνων για την αναγνώριση δεξαμενών σε ένα δεύτερο σετ από εικόνες SAR. Για την επίτευξη των στόχων των εργασιών καθορίστηκαν εκ των προτέρων σετ χαρακτηριστικών σε διάφορες εικόνες και εν συνεχεία χρησιμοποιήθηκε ΓΠ προκειμένου να επιλεγούν σχετικά χαρακτηριστικά και μέθοδοι ανάλυσης αυτών. Τα

αποτελέσματα συγκριθήκαν στη συνέχεια με τα χαρακτηριστικά που είχαν επιλεγεί αρχικά ώστε να ερευνηθεί η αποτελεσματικότητα του ΓΠ.

Ο **Belpaeme** (1999), περιέγραψε τον τρόπο με τον οποίο οπτικοί ανιχνευτές χαρακτηριστικών (visual feature detectors) ξεκινώντας από απλά πρωτογενή αντικείμενα διαμορφώνονται σε χαρακτηριστικά υψηλότερου επιπέδου. Τα πρωτογενή αυτά αντικείμενα, μερικά από τα οποία είναι εμπνευσμένα από την οπτική επεξεργασία, συνδυάστηκαν με την χρήση ΓΠ ώστε να διαμορφωθεί μια ιεραρχική ροή προς τα εμπρός για την εξαγωγή χαρακτηριστικών. Τα δεδομένα εκπαίδευσης για τους ανιχνευτές χαρακτηριστικών αποτελούνταν από μια σειρά πραγματικών εικόνων, που περιείχαν αντικείμενα ή πρόσωπα. Τα αποτελέσματα έδειξαν πως κάθε ομάδα ανιχνευτών, αυτό - οργανωνόταν σε ένα σύνολο το οποίο ήταν σε θέση να επιστρέφει διανύσματα χαρακτηριστικών γνωρισμάτων για την διάκριση της εικόνας.

Οι **Howard et al.** (1999), μελέτησαν την αυτόματη ανίχνευση πλοίων σε χαμηλής ανάλυσης SAR εικόνες. Οι στόχοι του σχεδιασμού του ανιχνευτή ήταν να μεγιστοποιηθεί η ακρίβεια εντοπισμού σε πολλαπλές εικόνες, να ελαχιστοποιηθεί ο υπολογιστικός φόρτος κατά τη διάρκεια της επεξεργασίας εικόνας και να ελαχιστοποιηθεί η προσπάθεια κατά το στάδιο του σχεδιασμού. Τα αποτελέσματα από μια εκτεταμένη αριθμητική μελέτη έδειξαν ότι με μια νέα προσέγγιση, με τη χρήση ΓΠ, αναπτύσσονται με επιτυχία ανιχνευτές, που πληρούν τους προηγούμενες στόχους. Κάθε ανιχνευτής αντιπροσωπεύει ένα αλγεβρικό τύπο και με τον τρόπο αυτό οι αρχές της ανίχνευσης μπορούν να ανακαλυφθούν και να επαναχρησιμοποιηθούν. Αυτό ήταν ένα σημαντικό πλεονέκτημα σε σχέση με τεχνικές τεχνητής νοημοσύνης οι οποίες χρησιμοποιούσαν κυρίως πολύπλοκες παραστάσεις.

Οι **Zhang και Ciesielski** (1999), περιέγραψαν μια προσέγγιση χρήσης ΓΠ για προβλήματα ανίχνευσης αντικειμένων, στα οποία έπρεπε να εντοπιστούν οι θέσεις μικρών αντικειμένων πολλών κατηγοριών σε μεγάλες εικόνες. Το πρόγραμμα χρησιμοποιεί ένα σύνολο χαρακτηριστικών τα οποία υπολογίζονται από ένα τετραγωνικό πεδίο εισόδου αρκετά μεγάλο ώστε να περιέχει κάθε ένα από τα αντικείμενα ενδιαφέροντος και το οποίο διατρέχει τις εικόνες με μορφή ενός κινούμενου παραθύρου, με σκοπό να εντοπίσει τα αντικείμενα που αναζητούνται. Η συνάρτηση καταλληλότητας βασίζεται στο ποσοστό ανίχνευσης και στο ποσοστό λανθασμένων εντοπισμών. Η μέθοδος δοκιμάστηκε σε τρία προβλήματα ανίχνευσης με αυξανόμενη δυσκολία και με τέσσερις διαφορετικές κλάσεις αντικειμένων ενδιαφέροντος. Στις εικόνες εύκολης και μέτριας δυσκολίας όλα τα αντικείμενα εντοπίστηκαν ορθά. Στις εικόνες με μεγαλύτερη δυσκολία υπήρξε σημαντικός αριθμός σφαλμάτων, ωστόσο τα αποτελέσματα ήταν σημαντικά καλύτερα από εκείνα που προέκυψαν από ένα πρόγραμμα βασισμένο σε νευρωνικό δίκτυο με τα ίδια προβλήματα.

Οι **Bhanu και Lin** (2002), μελέτησαν τον τρόπο να ανακαλύπτουν σύνθετες πράξεις και χαρακτηριστικά τα οποία εξελίσσονται από συνδυασμούς πρωτογενών πράξεων επεξεργασίας εικόνων, με σκοπό την εξαγωγή περιοχών ενδιαφέροντος από τις εικόνες αυτές. Η εργασία αυτή βασίστηκε στον ΓΠ. Το κίνητρο για την χρησιμοποίηση του ΓΠ ήταν ότι μέσω αυτού επιτυγχάνονται παρά πολύ συνδυασμοί πρωταρχικών πράξεων συμβατικοί και μη, ενώ η ανθρώπινη εξειδίκευση περιορίζεται λόγω της εμπειρίας, γνώσης και χρόνου, γεγονός που οδηγεί σε περιορισμένο αριθμό συμβατικών συνδυασμών μόνο. Μέσω των πειραματικών αποτελεσμάτων απέδειξαν ότι ο ΓΠ μπορεί να εντοπίσει πολύ σωστές σύνθετες πράξεις, αποτελούμενες από πρωτογενείς πράξεις, μέσω των οποίων

μπορούν να εξαχθούν αποτελεσματικά περιοχές ενδιαφέροντος σε παρόμοιες εικόνες με τα δεδομένα εκπαίδευσης.

Οι **Jeon et al.** (2002), παρουσίασαν μια τεχνική για την ανίχνευση των δρόμων σε μια εικόνα ραντάρ συνθετικού διαφράγματος (SAR) με τη χρήση ΓΑ. Οι δρόμοι σε μια εικόνα SAR μπορούν να μοντελοποιηθούν ως καμπύλες γραμμές κάποιου πλάτους. Τα τμήματα της καμπύλης, τα οποία αντιπροσωπεύουν τις υποψήφιες θέσεις για τους δρόμους, εξαγονται από την εικόνα χρησιμοποιώντας μία καμπυλόγραμμη δομή ανίχνευσης, και οι δρόμοι ανιχνεύονται με ακρίβεια ομαδοποιώντας τα τμήματα αυτά. Για το σκοπό αυτό, σχεδίασαν μια μέθοδο ομαδοποίησης των τμημάτων αυτών βασισμένη στην χρήση ΓΑ. Συνδύασαν τους αντιληπτικούς παράγοντες ομαδοποίησης με την χρήση ΓΑ και προσπάθησαν να μειώσουν το συνολικό υπολογιστικό κόστος της με την εισαγωγή της έννοιας της αναπτυσσόμενης περιοχής. Στη διαδικασία αυτή, μια αρχικά επιλεγμένη αρχικοποίηση εξελίσσεται σε ένα ομαδοποιημένο τμήμα μέσα από την επαναληπτική διαδικασία του ΓΑ, το οποίο και εξετάζει τμήματα μόνο μέσα σε μια επιλεγμένη περιοχή αναζήτησης. Για τον εντοπισμό των δρόμων με μεγαλύτερη ακρίβεια, μετά την επεξεργασία, αφαιρούνται τυχόν υπάρχοντα θορυβώδη τμήματα καμπυλών. Η μέθοδος αυτή εφαρμόστηκε στις ERS-1 και SIR-C/X SAR εικόνες οι οποίες έχουν ανάλυση περίπου 30 μέτρα. Τα πειραματικά αποτελέσματα έδειξαν ότι η μέθοδος αυτή μπορεί να ανιχνεύσει με ακρίβεια το οδικό δίκτυο καθώς και μεμονωμένα τμήματα δρόμων.

Οι **Wen-Bing et al.** (2003), παρουσίασαν μια μέθοδο με τρία επίπεδα κατωφλίωσης για την κατάτμηση εικόνας, βασισμένη στην πιθανότητα διαμερισματοποίησης, στην πιθανότητα ασαφούς λογικής και στην θεωρία της εντροπίας. Κάθε εικόνα χωρίζεται στο σκούρο, στο γκρι και στο λευκό τμήμα της, των οποίων συναρτήσεις για της ασαφής περιοχή είναι οι Z - function, Π- function και S-function. Υπάρχουν έξι παράμετροι σε αυτές τις συναρτήσεις που καθορίζουν το εύρος και τα χαρακτηριστικά της ασαφούς περιοχής. Η ασαφής περιοχή εντοπίζεται μέσω ΓΑ, με βάση την αρχή της μέγιστης ασαφούς εντροπίας. Η εικόνα κρατάει όσο το δυνατόν περισσότερη πληροφορία, όταν μετασχηματίζεται από το πεδίο της έντασης στο πεδίο της ασαφούς λογικής. Η διαδικασία για την εύρεση του βέλτιστου συνδυασμού των παραμέτρων της ασαφούς διαδικασία υλοποιείται με γενετικό αλγόριθμο και κατάλληλη κωδικοποίηση ώστε να αποφευχθούν χρωμοσώματα που δεν βελτιστοποιούν τις παραμέτρους. Τα πειραματικά αποτελέσματα δείχνουν ότι η προτεινόμενη μέθοδος αποδίδει καλή απόδοση.

Οι **Bhanu και Lin** (2004), δημοσίευσαν μια εργασία στην οποία χρησιμοποίησαν ΓΠ σε εικόνες SAR, IR και RGB. Στόχος τους ήταν να αυτοματοποιήσουν το σχεδιασμό ενός συστήματος ανίχνευσης αντικειμένων από εικόνες, με την αυτόματη σύνθεση αντικειμένων με διαδικασίες ανίχνευσης, χρησιμοποιώντας πρωτογενείς πράξεις και πρωτογενή χαρακτηριστικά. Λέγοντας πρωτογενείς πράξεις αναφέρονται είτε σε πράξεις μεταξύ δυο εικόνων όπως πρόσθεση αφαίρεση πολλαπλασιασμό κ.α. είτε σε πράξεις μέσα σε μια εικόνα όπως εφαρμογή μεγίστου, τυπικής απόκλισης, μέσου όρου, λογαρίθμου κ.α. σε ένα παράθυρο 3 x 3 ή 5 x 5 ή 7 x 7. Επιπλέον με τον όρο πρωτογενή χαρακτηριστικά εννοούνται διάφορα πρωτογενή σχήματα όπως τετράγωνα ορθογώνια κύκλοι κ.α. Ουσιαστικά εφαρμόζοντας συνδυασμό πρωτογενών πράξεων στις αρχικές εικόνες και με την βοήθεια των πρωτογενών χαρακτηριστικών πραγματοποιείται σύνθεση αντικειμένων μέσω ΓΠ. Βασίστηκαν στο ότι πολλές βασικές πράξεις μπορούν να εφαρμοστούν σε εικόνες και οι τρόποι συνδυασμού αυτών των πράξεων είναι σχεδόν άπειροι. Επιπλέον ενώ η εξειδίκευση των ανθρώπων περιορίζεται από την εμπειρία τον χρόνο και την γνώση, με αποτέλεσμα να μπορούν να πραγματοποιήσουν ένα μικρό μόνο αριθμό

συνδυασμών, οι ΓΑ, μπορούν να πραγματοποιήσουν τέτοιο πλήθος συνδυασμών, συμβατικών και μη, που ο άνθρωπος δεν μπορεί να φανταστεί καν. Μέσω των δοκιμών που έγιναν σε τμήματα των εικόνων, ώστε να βελτιωθεί η αποτελεσματικότητα της εκπαίδευσης, παρατηρήθηκε ότι οι ΓΑ μπορούσαν να συνθέσουν αποτελεσματικά σύνθετες πράξεις από πρωτογενείς και πρωτογενή χαρακτηριστικά έτσι ώστε να επιτευχθεί ανίχνευση αντικειμένων σε εικόνες, ενώ οι σύνθετες πράξεις μπορούσαν στην συνέχεια να εφαρμοστούν σε ολόκληρη την εικόνα εκπαίδευσης και σε άλλες παρόμοιες εικόνες δοκιμής.

Οι **Tseng et al.** (2007), παρουσίασαν ένα σύστημα ταξινόμησης βασισμένο σε κανόνες το οποίο προέρχεται από μια βελτιωμένη προσέγγιση ενός γενετικού αλγορίθμου και έχει ως σκοπό την ταξινόμηση της εδαφοκάλυψης δορυφορικών εικόνων σε κατηγορίες γης. Η ταξινόμηση της εδαφοκάλυψης σε κατηγορίες, χρησιμοποιούσε μέχρι πρόσφατα στατιστικές μεθόδους όπως η μέθοδος της μεγίστης πιθανοφάνειας. Ωστόσο τα τελευταία χρόνια έχει υπάρξει πλήθος μελετών που χρησιμοποιούν τεχνικές τεχνητής νοημοσύνης προκειμένου να πραγματοποιηθεί η ταξινόμηση αυτή. Ένα μειονέκτημα της τεχνικής αυτής είναι η δυσκολία υλοποίησης των τελικών κανόνων ταξινόμησης από τον χρήστη. Ο προτεινόμενος αλγόριθμος εφαρμόστηκε σε προβλήματα ταξινόμησης σε δυο διαφορετικά σύνολα εικόνων. Το πρώτο σύνολο εικόνων αφορούσε δορυφορικές εικόνες Landsat. Τα δεδομένα διαχωρίστηκαν σε δεκαπέντε διαφορετικές κατηγορίες εδάφους ενώ κάθε κατηγορία περιείχε έξι περίπου διαφορετικές περιπτώσεις. Μέσω του ΓΠ δημιουργήθηκε ένα σύνολο από δεκαπέντε κανόνες οι οποίοι δοκιμάστηκαν σε εβδομήντα πέντε περιπτώσεις, στις εβδομήντα τέσσερις από τις οποίες έδωσαν σωστά αποτελέσματα ταξινόμησης. Το δεύτερο σύνολο εικόνων αφορούσε εικόνες AVIRIS. Τα δεδομένα διαχωρίστηκαν σε έξι διαφορετικές κατηγορίες εδάφους. Μέσω του ΓΠ δημιουργήθηκε ένα σύνολο από έξι κανόνες ενώ μέσω της τετριμμένης μεθόδου δημιουργίας κανόνων ένα σύνολο από 11 κανόνες, προκειμένου να πραγματοποιηθεί σύγκριση των αποτελεσμάτων. Το ποσοστό επιτυχίας της ταξινόμησης μέσω του δείκτη PRA έφτασε το 75,1% για το σύνολο κανόνων του ΓΠ και το 70,7% για το δεύτερο σύνολο κανόνων.

Οι **Kamal et al.** (2007), παρουσίασαν μια μέθοδο πολυεπίπεδης κατωφλίωσης η οποία επιτρέπει τον καθορισμό του πλήθους των κατωφλίων, καθώς και της τιμής για κάθε κατωφλίωση. Η μέθοδος αυτή συνδυάζει ένα ΓΑ με ένα μετασχηματισμό wavelet. Αρχικά το μήκος του ιστογράμματος μειώνεται με χρήση του μετασχηματισμού wavelet. Με βάση αυτή την εκδοχή του ιστογράμματος, υπολογίζονται με χρήση ΓΑ το πλήθος των κατωφλίων και οι τιμές κατωφλίωσης. Οι τιμές κατωφλίωσης εν συνεχεία προβάλλονται στον αρχικό χώρο, ενώ μια διαδικασία βελτίωσης προσθέτει περισσότερη ακρίβεια στις τιμές κατωφλίωσης. Τα παραπάνω εφαρμόζονται πειραματικά σε εικόνες και αποδεικνύουν την συγκεκριμένη μέθοδο.

Οι **Santos et al.** (2010), παρουσίασαν μια διαδραστική τεχνική για την ταξινόμηση εικόνων τηλεπισκόπησης. Οι χρήστες μπορούν να αλληλεπιδρούν με το σύστημα ταξινόμησης, επισημαίνοντας τις περιοχές ενδιαφέροντος και μη. Η ανατροφοδότηση αυτή των πληροφοριών συνδέεται με την εφαρμογή ΓΠ η οποία έχει ως στόχο την εκμάθηση των προτιμήσεων των χρηστών συνδυάζοντας περιγραφές των περιοχών της εικόνας οι οποίες και κωδικοποιούνται μέσω των φασματικών ιδιοτήτων και της υψής. Τα πειράματα έδειξαν ότι η προτεινόμενη μέθοδος είναι αποτελεσματική στην ταξινόμηση των εικόνων και υπερτερεί της παραδοσιακής μεθόδου MaxVer.

3. ΜΕΘΟΔΟΛΟΓΙΑ

3.1 Γενικά

Στο κεφάλαιο αυτό θα παρουσιαστούν:

- Η ανάλυση του **αλγορίθμου ελέγχου της ακρίβειας** μιας κατάτμησης με βάση προϋπάρχοντες δείκτες καθώς και με βάση τους δείκτες που πρότειναν οι Yong Liu et al (2012),
- Η ανάλυση του **ΓΑ**,
- Οι εικόνες που **χρησιμοποιήθηκαν** στη εφαρμογή του ΓΑ,
- Ο **τρόπος δημιουργίας** των εικόνων αναφοράς,
- Η ανάλυση **ευαισθησίας** και επιλογή **παραμέτρων** με βάση την λειτουργία του ΓΑ.

3.2 Ανάλυση του Υλοποιημένου Αλγορίθμου Ελέγχου Ακρίβειας της Κατάτμησης καθώς και του Γενετικού Αλγορίθμου

3.2.1 Γενικά

Το πρόβλημα για την επίλυση του οποίου χρησιμοποιούνται οι αλγόριθμοι κατάτμησης που αναπτύχθηκαν στο εδάφιο 2.4, χαρακτηρίζεται ως συνδυαστικό καθώς εμπλέκει επαναληπτικές διαδικασίες των αλγορίθμων κατάτμησης, έλεγχο της ακρίβειας της κάθε κατάτμησης μέσω του αλγορίθμου ελέγχου και εφαρμογή του ΓΑ για επίτευξη της εύρεσης των βέλτιστων παραμέτρων ώστε να υλοποιηθεί η αντίστοιχη βέλτιστη κατάτμηση. Ουσιαστικά ο αλγόριθμος ελέγχου της ακρίβειας της κατάτμησης σε συνδυασμό με τον αλγόριθμο κατάτμησης που χρησιμοποιείται κάθε φορά, αποτελεί την συνάρτηση αξιολόγησης του ΓΑ, την οποία και καλεί κάθε φορά ώστε να υλοποιηθεί. Για την υλοποίηση των αλγορίθμων αρχικά εισάγονται οι απαραίτητες βιβλιοθήκες της γλώσσας Python που απαιτούνται κάθε φορά.

3.2.2 Ανάλυση του αλγορίθμου ελέγχου της ακρίβειας της κατάτμησης βασισμένη

Ο αλγόριθμος ελέγχου της ακρίβειας ή αλλιώς συνάρτηση αξιολόγησης βασίστηκε στην εργασία των Yong Liu et al (2012). Συγκεκριμένα μέσω της εργασίας αυτής παρουσιάζονται τέσσερις υπάρχοντες δείκτες για την αξιολόγηση μιας κατάτμησης και προτείνονται τρεις νέοι δείκτες. Οι υπάρχοντες δείκτες βασίζονται στην γεωμετρικές σχέσεις που μπορεί να έχει ένα πολύγωνο αναφοράς με ένα αντιστοιχισμένο πολύγωνο κατάτμησης ενώ οι προτεινόμενοι δείκτες βασίζονται επιπλέον και στις αριθμητικές μεταξύ τους σχέσεις. Οι υπάρχοντες δείκτες προτάθηκαν στην εργασία των Weidner (2008) και Clinton et al.(2010). Συγκεκριμένα οι υπάρχοντες δείκτες είναι οι:

- Quality Rate (QR),
- Over segmentation Rate (OR),
- Under segmentation Rate (UR) και
- Euclidian Distance 1 (ED1),

ενώ οι προτεινόμενοι είναι:

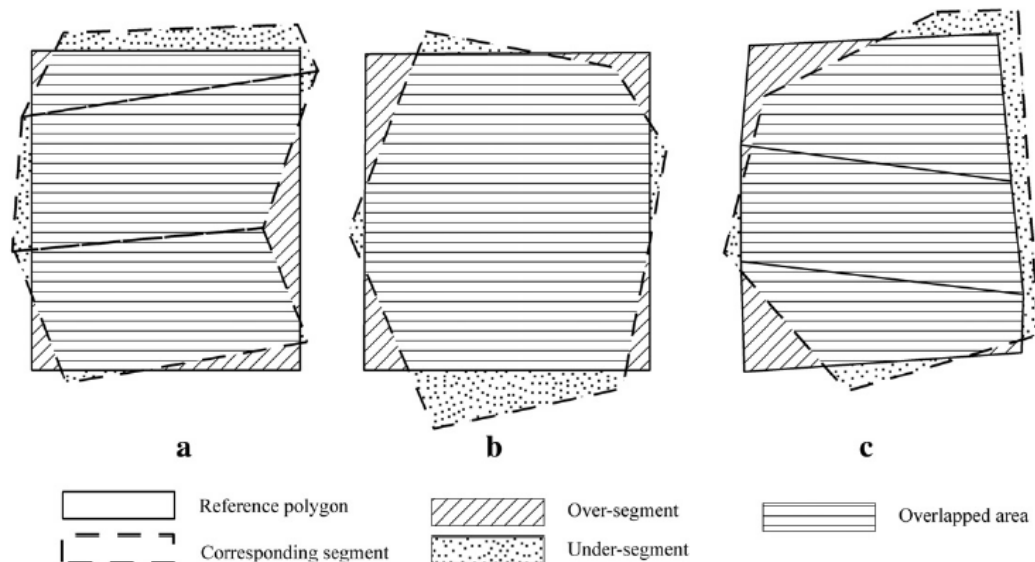
- Potential Segmentation Error (PSE),
- Number-of-Segments Ratio (NSR) και
- Euclidian Distance 2 (ED2).

Με τον όρο αντιστοιχισμένο πολύγωνο κατάτμησης (corresponding segment) εννοείται ένα πολύγωνο κατάτμησης που προκύπτει από την διαδικασία της κατάτμησης, το οποίο όμως πληροί ένα από τα εξής δυο κριτήρια. Η τομή μεταξύ του πολυγώνου κατάτμησης και του πολυγώνου αναφοράς πρέπει να είναι μεγαλύτερη ή ίση είτε από το μισό του πολυγώνου κατάτμησης είτε από το μισό του πολυγώνου αναφοράς. Σε περίπτωση που δεν υλοποιείται ένα από τα παραπάνω δυο κριτήρια τότε το πολύγωνο κατάτμησης δεν λαμβάνει τον χαρακτηρισμό αντιστοιχισμένο και δεν χρησιμοποιείται στις διαδικασίες υπολογισμού των δεικτών ελέγχου της ακρίβειας της κατάτμησης.

Οι γεωμετρικές σχέσεις μεταξύ πολυγώνων αναφοράς και αντιστοιχισμένων πολυγώνων κατάτμησης είναι οι εξής:

- Επικάλυψη (overlap): Το επικαλυπτόμενο τμήμα ορίζεται το κοινό τμήμα μεταξύ πολυγώνου αναφοράς και αντιστοιχισμένου πολυγώνου κατάτμησης,
- Over - segment: Είναι το τμήμα του πολυγώνου αναφοράς το οποίο δεν επικαλύπτεται από το αντιστοιχισμένο πολύγωνο κατάτμησης και
- Under - segment: Είναι το τμήμα του αντιστοιχισμένου πολυγώνου κατάτμησης το οποίο βρίσκεται εκτός του πολυγώνου αναφοράς.

Στην εικόνα 3.1 παρουσιάζονται εποπτικά οι παραπάνω σχέσεις.



Εικόνα 3.1 - Γεωμετρικές σχέσεις μεταξύ πολυγώνων αναφοράς και πολυγώνων κατάτμησης

Οι αριθμητικές σχέσεις μεταξύ πολυγώνων αναφοράς και αντιστοιχισμένων πολυγώνων κατάτμησης είναι οι εξής:

- Ένα προς πολλά (one to many), όπου ένα πολύγωνο αναφοράς αντιστοιχεί το λιγότερο σε δυο αντιστοιχισμένα πολύγωνα κατάτμησης,
- Ένα προς ένα (one to one), όπου ένα πολύγωνο αναφοράς αντιστοιχεί σε ένα αντιστοιχισμένο πολύγωνο κατάτμησης

- Πολλά προς ένα (many to one), όπου τουλάχιστον δυο πολύγωνα αναφοράς αντιστοιχούν σε ένα αντιστοιχισμένο πολύγωνο κατάτμησης.

Οι ποσότητες που υπολογίζονται με βάση τις παραπάνω σχέσεις και οι οποίες λαμβάνουν μέρος στον υπολογισμό των δεικτών που αναφέρθηκαν είναι οι εξής:

- **Sum Overlapped:** πρόκειται για το σύνολο των αλληλοεπικαλυπτόμενων εικονοστοιχείων (τομή) μεταξύ των πολυγώνων αναφοράς και των αντιστοιχισμένων πολυγώνων κατάτμησης,
- **Sum Union:** πρόκειται για το σύνολο των εικονοστοιχείων που προκύπτουν από την ένωση των πολυγώνων αναφοράς και των αντιστοιχισμένων πολυγώνων κατάτμησης,
- **Sum Oversegment:** πρόκειται για το σύνολο των εικονοστοιχείων που ανήκουν σε πολύγωνο αναφοράς αλλά δεν ανήκουν σε αντιστοιχισμένο πολύγωνο κατάτμησης,
- **Sum Reference:** πρόκειται για το σύνολο των εικονοστοιχείων των πολυγώνων αναφοράς,
- **Sum Undersegment:** πρόκειται για το σύνολο των εικονοστοιχείων που ανήκουν σε αντιστοιχισμένα πολύγωνα κατάτμησης αλλά δεν ανήκουν σε πολύγωνο αναφοράς,
- **Sum Segment:** πρόκειται για το σύνολο των εικονοστοιχείων των αντιστοιχισμένων πολυγώνων κατάτμησης,
- **Ref max:** είναι το πλήθος των πολυγώνων αναφοράς και
- **Cor Obj:** είναι το πλήθος των πολυγώνων κατάτμησης.

Οι δείκτες που υπολογίζονται ως αποτέλεσμα των παραπάνω μεγεθών έχουν ως εξής:

$$\begin{aligned} \diamond \text{ QR} &= 1 - \frac{\text{sum_overlapped}}{\text{sum_union}} \\ \diamond \text{ OR} &= \frac{\text{sum_oversegment}}{\text{sum_reference}} \\ \diamond \text{ UR} &= \frac{\text{sum_undersegment}}{\text{sum_segment}} \\ \diamond \text{ ED1} &= \frac{\sqrt{\text{OR}^2 + \text{UR}^2}}{2} \\ \diamond \text{ PSE} &= \frac{\text{sum_undersegment}}{\text{sum_reference}} \\ \diamond \text{ NSR} &= \frac{|\text{ref_max} - \text{cor_obj}|}{\text{ref_max}} \text{ και} \\ \diamond \text{ ED2} &= \sqrt{\text{PSE}^2 + \text{NSR}^2} \end{aligned}$$

Οι τιμές που λαμβάνουν οι παραπάνω δείκτες καθώς και οι ερμηνείες των τιμών αυτών είναι οι εξής:

- **QR:** 0 - 1, όπου η τιμή 0 ερμηνεύεται ως η μη ύπαρξη ούτε over segment ούτε under segment εικονοστοιχείων δηλαδή η επιθυμητή γεωμετρική κατάτμηση, ενώ η τιμή 1 ερμηνεύεται ως ανεπιθύμητη κατάτμηση,
- **OR:** 0 - 1, όπου η τιμή 0 ερμηνεύεται ως η μη ύπαρξη over segment εικονοστοιχείων, ενώ η τιμή 1 ερμηνεύεται ως ανεπιθύμητη κατάτμηση,

- **UR:** 0 - 1, όπου η τιμή 0 ερμηνεύεται ως η μη ύπαρξη under segment εικονοστοιχείων, ενώ η τιμή 1 ερμηνεύεται ως ανεπιθύμητη κατάτμηση,
- **ED1:** 0 - 1, όπου η τιμή του εξαρτάται από τις τιμές των δεικτών OR και UR και ερμηνεύεται ως η απόσταση από την αρχή των αξόνων στον δισδιάστατο χώρο OR - UR,
- **PSE:** 0 - $+\infty$, όπου η τιμή 0 ερμηνεύεται ως η μη ύπαρξη under segment εικονοστοιχείων, ενώ όσο η τιμή αυξάνεται συνεπάγεται αύξηση των under segment τμημάτων,
- **NSR:** 0 - $+\infty$, όπου η τιμή 0 ερμηνεύεται ως η προτιμώμενη σχέση ένα προς ένα μεταξύ αντικειμένων αναφοράς και αντιστοιχισμένων αντικειμένων κατάτμησης, ενώ μια μεγάλη τιμή συνεπάγεται την δημιουργία της σχέσης ένα προς πολλά μεταξύ των αντικειμένων. Ο δείκτης δίνεται μέγιστος όταν όλα τα αντιστοιχισμένα πολύγωνα κατάτμησης αποτελούνται από ένα εικονοστοιχείο το καθένα και
- **ED2:** 0 - $+\infty$, όπου η τιμή του εξαρτάται από τις τιμές των δεικτών PSE και NSR και ερμηνεύεται ως η απόσταση από την αρχή των αξόνων στον δισδιάστατο χώρο PSE - NSR. Τιμή 0 ερμηνεύεται ως το επιθυμητό γεωμετρικό και αριθμητικό ταίριασμα μεταξύ των πολυγώνων αναφοράς και κατάτμησης ενώ μεγάλη τιμή του δείκτη υποδεικνύει είτε σημαντική γεωμετρική ή αριθμητική απόκλιση είτε και τις δυο.

Η ανάπτυξη του αλγορίθμου ελέγχου της ακρίβειας της κατάτμησης, όπως και του ΓΑ υλοποιήθηκε στην γλώσσα python. Αναλύεται στα εξής τμήματα:

ΤΜΗΜΑ Α: Προπαρασκευαστικές ενέργειες για την επιλογή εικόνας

```
try:
    from osgeo import gdal
    from osgeo.gdalconst import *
except ImportError:
    import gdal
    from gdalconst import *

try:
    import numpy
except ImportError:
    import Numeric as numpy

import sys
import cv2
from cv2 import cv
import math
import scipy
import pylab
import matplotlib
from scipy import ndimage
from scipy import misc
```

Στο τμήμα αυτό του αλγορίθμου εισάγονται οι απαραίτητες βιβλιοθήκες ώστε να μπορέσει να υλοποιηθεί η επίτευξη της δημιουργίας του αλγορίθμου. Οι βιβλιοθήκες αυτές έχουν προηγουμένως εγκατασταθεί στο λειτουργικό ώστε να μπορέσουν να χρησιμοποιηθούν.

ΤΜΗΜΑ Β: Εισαγωγή των εικόνων κατάτμησης και αναφοράς και υλοποίηση ελέγχων σε αυτές

```

def Usage():
    print('Usage:diplomatiki_RS.py segmentation_file reference_file')
    print('')
    sys.exit( 1 )

segfile = 'segment_image.tif'
reffile = 'reference_image.tif'

# Parse command line arguments.
i = 1
while i < len(sys.argv):
    arg = sys.argv[i]

    if segfile is None:
        segfile = arg

    elif reffile is None:
        reffile = arg
    else:
        Usage()
    i = i + 1
if segfile is None:
    Usage()
if reffile is None:
    Usage()
segdataset = gdal.Open( segfile, GA_ReadOnly )
refdataset = gdal.Open( reffile, GA_ReadOnly )
if (segdataset.RasterCount != 1):
    print('Error: Segmentation file has more than 1 band')
    sys.exit( 1 )
if (refdataset.RasterCount != 1):
    print('Error: Ground truth file has more than 1 band')
    sys.exit( 1 )
if ((segdataset.RasterXSize != refdataset.RasterXSize) or
    (segdataset.RasterYSize != refdataset.RasterYSize)):
    print('Error: Segmentation and Ground truth file have different
    dimensions')
    sys.exit( 1 )

```

Στο τμήμα αυτό αρχικά υλοποιείται συνάρτηση ώστε ο χρήστης εάν επιθυμεί να εκτελέσει την εφαρμογή μέσω του παραθύρου cmd. Ακολουθεί έλεγχος για το εάν οι μεταβλητές «segfile» και «reffile» αντιστοιχούν στις εικόνες κατάτμησης και αναφοράς αντίστοιχα. Εάν αυτό δεν συμβαίνει ο αλγόριθμος τερματίζει. Επόμενος έλεγχος είναι η διαπίστωση για το πόσα κανάλια έχουν οι εικόνες κατάτμησης και αναφοράς. Εάν η τιμή είναι διαφορετική του ενός, δηλαδή υπάρχουν περισσότερα από ένα κανάλια, τότε ο αλγόριθμος τερματίζει καθώς οι εικόνες κατάτμησης και αναφοράς πρέπει να έχουν ένα και μόνο κανάλι. Ο τελευταίος έλεγχος αφορά την σύγκριση των διαστάσεων των εικόνων αναφοράς και κατάτμησης. Όμοια με πριν εάν οι διαστάσεις των δυο εικόνων είναι διαφορετικές ο αλγόριθμος τερματίζει.

ΤΜΗΜΑ Γ: Δημιουργία λίστας με τις μοναδικές τιμές των εικονοστοιχείων των πολυγώνων κατάτμησης που βρίσκονται εντός των πολυγώνων αναφοράς

```

segband = segdataset.GetRasterBand(1)
refband = refdataset.GetRasterBand(1)
(segmin, segmax) = segband.ComputeRasterMinMax()
(refmin, refmax) = refband.ComputeRasterMinMax()
list_val_of_pixel=[]
scanline_cl = segband.ReadAsArray(0, 0, segband.XSize, segband.YSize)
scanline_tr = refband.ReadAsArray(0, 0, refband.XSize, refband.YSize)
z=0
for i in range(int(refmin+1), int(refmax+1)):
    for y in range(0, segband.XSize):
        for x in range(0, segband.YSize):
            if scanline_tr[x][y]==i:
                list_val_of_pixel.append(scanline_cl[x][y])
                for k in range(z,len(list_val_of_pixel)-1):
                    if list_val_of_pixel[len(list_val_of_pixel)-1]==list_val_of_pixel[k]:
                        list_val_of_pixel.pop()
                        break
                list_val_of_pixel.append((-1))
            z=len(list_val_of_pixel)-1

```

Στο τμήμα αυτό αρχικά υπολογίζονται τα στατιστικά για τις εικόνες αναφοράς και κατάτμησης. Ουσιαστικά υπολογίζεται το πλήθος των εικονοστοιχείων σε αυτές στις δυο διαστάσεις και επιπλέον η μικρότερη και η μεγαλύτερη χαρακτηριστική τιμή που έχουν τα αντικείμενα τόσο στην εικόνα κατάτμησης όσο και στην εικόνα αναφοράς. Δηλαδή για παράδειγμα σε μια εικόνα κατάτμησης με 1932 αντικείμενα και σε μια εικόνα αναφοράς με 3 αντικείμενα αναφοράς και με διαστάσεις η κάθε μια 300 pixel x 200 pixel το αποτέλεσμα του κώδικα θα είναι:

```

Minimum value of segmentation object: 1.0
Maximum value of segmentation object: 1932.0
Minimum value of reference object: 0.0
Maximum value of reference object: 3.0
Number of Pixel Y in segmentation image : 300
Number of Pixel X in segmentation image : 200
Number of Pixel Y in reference image : 300
Number of Pixel X in reference image : 200

```

Ο απώτερος στόχος του παραπάνω κώδικα είναι να δημιουργηθεί μια λίστα στην οποία να περιέχονται οι μοναδικές τιμές των εικονοστοιχείων των πολυγώνων κατάτμησης της εικόνας κατάτμησης οι οποίες περιέχονται εντός των πολυγώνων αναφοράς της εικόνας αναφοράς. Για το λόγο αυτό για κάθε πολύγωνο αναφοράς διατρέχετε η εικόνα κατάτμησης. Εντοπίζονται οι τιμές των εικονοστοιχείων των πολυγώνων κατάτμησης τα οποία βρίσκονται εντός του συγκεκριμένου κάθε φορά, πολυγώνου αναφοράς. Κάθε τιμή αποθηκεύεται μια μόνο φορά μέσα στην λίστα, μετά από έλεγχο που γίνεται ώστε να διαγράφεται η τιμή εάν εντοπιστεί ξανά μέσα στην λίστα. Μετά το τέλος του ελέγχου για κάθε πολύγωνο αναφοράς προστίθεται η αρνητική τιμή «-1», ώστε να διακρίνονται μεταξύ τους οι τιμές για κάθε πολύγωνο αναφοράς. Η τιμή επιλέχτηκε ώστε να μην δημιουργήσει πρόβλημα στον αλγόριθμο καθώς όλες οι τιμές που αποθηκεύονται στην λίστα είναι θετικές ή μηδέν. Το αποτέλεσμα που προκύπτει ως συνέχεια του παραπάνω είναι:

Oi monadikes times ton pixel pou vriskontai pano apo ta antikeimena anaforas, diaxorismena me tin timi -1 einai:

[1515, 1389, 1390, 1447, 1361, 1331, 1468, 1427, 1595, 1489, -1, 1866, 1821, 1675, 1878, 1843, 1844, 1595, -1, 1105, 768, 713, 598, 1237, 1306, 1126, 734, 677, 638, 729, -1]

ΤΜΗΜΑ Δ: Υπολογισμός του ποσοστού των εικονοστοιχείων των πολυγώνων κατάτμησης που βρίσκονται εντός εκάστου πολυγώνου αναφοράς σε σχέση με το πλήθος των εικονοστοιχείων του αντίστοιχου πολυγώνου κατάτμησης αλλά και σε σχέση με το πλήθος των εικονοστοιχείων του αντίστοιχου πολυγώνου αναφοράς

```
array_pixel_seg = numpy.zeros((len(list_val_of_pixel),6))
array_pixel_ref = numpy.zeros((int(refmax+1),2))

for refmax in range (0,int(refmax+1)):
    array_pixel_ref[refmax][0]=refmax
    for y in range(0, segband.XSize):
        for x in range(0, segband.YSize):
            if scanline_tr[x][y]==refmax:
                array_pixel_ref[refmax][1] = array_pixel_ref
[refmax][1] + 1

i=1
for k in range(0,len(list_val_of_pixel)):
    array_pixel_seg[k][0]=list_val_of_pixel[k]
    if array_pixel_seg[k][0]==-1:
        i=i+1
    for y in range(0, segband.XSize):
        for x in range(0, segband.YSize):
            if (list_val_of_pixel[k]==scanline_cl[x][y]) and
(scanline_tr[x][y]==i):
                array_pixel_seg[k][1]=array_pixel_seg[k][1]+1
            elif (list_val_of_pixel[k]==scanline_cl[x][y]) and
(scanline_tr[x][y]!=i):
                array_pixel_seg[k][2]=array_pixel_seg[k][2]+1
                if list_val_of_pixel[k]==scanline_cl[x][y]:
                    array_pixel_seg[k][3]=array_pixel_seg[k][3]+1
            if (array_pixel_seg[k][0])!=(-1):
                array_pixel_seg[k][4]=(array_pixel_seg[k][1])/(array_pixel_seg
m[k][3])

i=1
for k in range(0,len(list_val_of_pixel)):
    if array_pixel_seg[k][0]==-1:
        i=i+1# gia na parei ola ta antikeimena anaforas
    if (array_pixel_seg[k][0])!=(-1):
        array_pixel_seg[k][5]=(array_pixel_seg[k][1])/(array_pixel_ref
[i][1])
```

Στο τμήμα αυτό αρχικά δημιουργούνται δυο πίνακες. Στον πρώτο πίνακα αποθηκεύονται, για κάθε πολύγωνο κατάτμησης, που σε οποιοδήποτε ποσοστό βρίσκεται εντός πολυγώνου αναφοράς, ο κωδικός του πολυγώνου, το πλήθος των εικονοστοιχείων εντός του πολυγώνου αναφοράς, το πλήθος των εικονοστοιχείων εκτός, το σύνολο των εικονοστοιχείων του πολυγώνου κατάτμησης, το ποσοστό των εικονοστοιχείων εντός σε σχέση με το σύνολο των εικονοστοιχείων του πολυγώνου κατάτμησης και το ποσοστό των εικονοστοιχείων εντός σε σχέση με το σύνολο των εικονοστοιχείων του πολυγώνου αναφοράς. Στον δεύτερο πίνακα αποθηκεύεται το πλήθος των εικονοστοιχείων κάθε πολυγώνου αναφοράς, πληροφορία που χρησιμοποιείται για να συμπληρωθεί η τελευταία στήλη του πρώτου πίνακα.

Με την εκτέλεση του κώδικα αρχικά υπολογίζονται για κάθε πολύγωνο αναφοράς το πλήθος των εικονοστοιχείων και αποθηκεύονται στον πίνακα με όνομα «array_pixel_ref». Ακολουθεί ο υπολογισμός των τιμών του δεύτερου πίνακα. Για να ελεγχθούν τα πολύγωνα κατάτμησης που σε οποιοδήποτε ποσοστό βρίσκονται εντός πολυγώνου αναφοράς χρησιμοποιείται η λίστα «list_val_of_pixel» που δημιουργήθηκε στο Τμήμα Γ και η οποία περιείχε ανά πολύγωνο αναφοράς τους μοναδικούς κωδικούς των πολυγώνων κατάτμησης που βρίσκονταν σε οποιοδήποτε ποσοστό εντός αυτού. Στο τέλος υπολογίζεται η τελευταία στήλη του δεύτερου πίνακα χρησιμοποιώντας τα αποτελέσματα της δεύτερης στήλης αυτού και της δεύτερης στήλης του πρώτου πίνακα. Το τελικό αποτέλεσμα που προκύπτει ως συνέχεια του τμήματος Γ είναι:

To plithos ton pixel ana timi antikeimenou anaforas einai:

```
[[ 0. 53498.]
 [ 1. 1300.]
 [ 2. 763.]
 [ 3. 4439.]]
```

value :Value of pixel in segmentation image

nopi :Number of segmentation pixel in reference object

nopo :Number of segmentation pixel out of reference object

sum :Sum of pixel of segment objects

sir :Percent of segment objects which is inside reference objects

ris :Percent of reference objects which is inside segment objects

****Diaxorismos ton antikeimenon anaforas me tin grammi pou periexei -1****

<i>value</i>	<i>nopi</i>	<i>nopo</i>	<i>sum</i>	<i>sir</i>	<i>ris</i>
[[1515.	21.	311.	332.	0.063	0.016]
[1389.	182.	21.	203.	0.897	0.14]
[1390.	134.	1.	135.	0.993	0.103]
[1447.	210.	0.	210.	1.	0.162]
[1361.	3.	123.	126.	0.024	0.002]
[1331.	289.	116.	405.	0.714	0.222]
[1468.	208.	38.	246.	0.846	0.16]
[1427.	242.	7.	249.	0.972	0.186]
[1595.	5.	2000.	2005.	0.002	0.004]
[1489.	6.	122.	128.	0.047	0.005]
[-1.	0.	0.	0.	0.	0.]

```
[ 1866.   3.  122.  125.   0.024  0.004]
[ 1821.  114.   65.  179.   0.637  0.149]
[ 1675.   74.  273.  347.   0.213  0.097]
[ 1878.  148.  255.  403.   0.367  0.194]
[ 1843.  272.   1.  273.   0.996  0.356]
[ 1844.  118.   0.  118.   1.   0.155]
[ 1595.   34. 1971. 2005.   0.017  0.045]
[  -1.   0.   0.   0.   0.   0. ]
[ 1105.   13.  117.  130.   0.1   0.003]
[  768.   25. 2995. 3020.   0.008  0.006]
[  713. 3432.   5. 3437.   0.999  0.773]
[  598.   4.  653.  657.   0.006  0.001]
[ 1237.   5.  132.  137.   0.036  0.001]
[ 1306.   9.  331.  340.   0.026  0.002]
[ 1126.  287.  13.  300.   0.957  0.065]
[  734.  514.  58.  572.   0.899  0.116]
[  677.  146.   0.  146.   1.   0.033]
[  638.   2.  299.  301.   0.007  0. ]
[  729.   2.  110.  112.   0.018  0. ]
[  -1.   0.   0.   0.   0.   0. ]]
```

ΤΜΗΜΑ Ε: Υπολογισμός των απαραίτητων στοιχείων για υλοποίηση του ελέγχου ακρίβειας της κατάτμησης και αποθήκευση τους σε πίνακα

```
results_pixel = numpy.zeros((int(refmax),7))
ref_obj=1
print      '*****', 'Reference      Object:
',ref_obj,'*****'

for i in range(0,(len(list_val_of_pixel)-1)):
    results_pixel[ref_obj-1][0]=ref_obj
    if array_pixel_segm[i][3]==0:
        ref_obj=ref_obj+1
        print      '\n\n*****', 'Reference      Object:
',ref_obj,'*****'

        if array_pixel_segm[i][3]!=0:
            if      (array_pixel_segm[i][4]>=0.5      or
array_pixel_segm[i][5]>=0.5):
                results_pixel[ref_obj-1][1]=results_pixel[ref_obj-
1][1]+array_pixel_segm[i][1]
                print      '%.0f'%array_pixel_segm[i][0],':Corresponding
object','--total pixel in common:',
                print      '%.0f'%array_pixel_segm[i][1],      of
', '%.0f'%array_pixel_segm[i][3], 'or', '%.2f'%(array_pixel_segm[i][4]*100)
, '%'

                if array_pixel_segm[i][5]>=0.5:
                    print ' and total pixel percent from reference to
segment object %.2f'%(array_pixel_segm[i][5]*100), '%'

                results_pixel[ref_obj-1][5]=results_pixel[ref_obj-
```

```

1][5]+1
        results_pixel[ref_obj-1][3]=results_pixel[ref_obj-
1][3]+array_pixel_segm[i][2]

    else :
        results_pixel[ref_obj-1][2]=results_pixel[ref_obj-
1][2]+array_pixel_segm[i][1]
        print    '%.0f'%array_pixel_segm[i][0],':No    Corresponding
object','--total pixel in common:', '%.0f'%array_pixel_segm[i][1],
        print
        , '%.0f'%array_pixel_segm[i][3], 'or', '%.2f'% (array_pixel_segm[i][4]*100)
        , '%
        results_pixel[ref_obj-1][6]=results_pixel[ref_obj-1][6]+1

        results_pixel[ref_obj-1][4]=results_pixel[ref_obj-
1][3]+results_pixel[ref_obj-1][2]+results_pixel[ref_obj-1][1]

print results_pixel

```

Στο τμήμα αυτό δημιουργείται ένας νέος πίνακας. Στον πίνακα αυτό αποθηκεύονται, για κάθε πολύγωνο αναφοράς, οι πληροφορίες των πολυγώνων κατάτμησης που σε οποιοδήποτε ποσοστό περιέχονται μέσα στο πολύγωνο αναφοράς. Συγκεκριμένα αποθηκεύονται ο κωδικός αριθμός του πολυγώνου αναφοράς, το πλήθος των κοινών εικονοστοιχείων ανάμεσα στο πολύγωνο αναφοράς και στα πολύγωνα κατάτμησης που περιέχονται σε αυτό είτε τμηματικά είτε συνολικά (overlapped area), το πλήθος των εικονοστοιχείων που ανήκουν στο πολύγωνο αναφοράς αλλά όχι στα πολύγωνα κατάτμησης (over-segment) και το πλήθος των εικονοστοιχείων που ανήκουν στα πολύγωνα κατάτμησης αλλά όχι στο πολύγωνο αναφοράς (under-segment). Επιπλέον αποθηκεύεται το σύνολο των παραπάνω εικονοστοιχείων όλων των κατηγοριών καθώς και το πλήθος των πολυγώνων κατάτμησης που λαμβάνουν μέρος στους παρακάτω υπολογισμούς (corresponding segment). Για να χαρακτηριστεί ένα υποψήφιο πολύγωνο κατάτμησης ως αντιστοιχισμένο πρέπει η τομή του με το πολύγωνο αναφοράς να υπερβαίνει το 50% του συνόλου των εικονοστοιχείων είτε του πολυγώνου αναφοράς είτε του πολυγώνου κατάτμησης. Το τελικό αποτέλεσμα που προκύπτει ως συνέχεια του τμήματος Δ είναι:

```

***** Reference Object: 1 *****
1515 :No Corresponding object --total pixel in common: 21 of 332 or 6.33 %
1389 :Corresponding object --total pixel in common: 182 of 203 or 89.66 %
1390 :Corresponding object --total pixel in common: 134 of 135 or 99.26 %
1447 :Corresponding object --total pixel in common: 210 of 210 or 100.00 %
1361 :No Corresponding object --total pixel in common: 3 of 126 or 2.38 %
1331 :Corresponding object --total pixel in common: 289 of 405 or 71.36 %
1468 :Corresponding object --total pixel in common: 208 of 246 or 84.55 %
1427 :Corresponding object --total pixel in common: 242 of 249 or 97.19 %
1595 :No Corresponding object --total pixel in common: 5 of 2005 or 0.25 %
1489 :No Corresponding object --total pixel in common: 6 of 128 or 4.69 %
***** Reference Object: 2 *****
1866 :No Corresponding object --total pixel in common: 3 of 125 or 2.40 %
1821 :Corresponding object --total pixel in common: 114 of 179 or 63.69 %

```

```

1675 :No Corresponding object --total pixel in common: 74 of 347 or 21.33 %
1878 :No Corresponding object --total pixel in common: 148 of 403 or 36.72 %
1843 :Corresponding object --total pixel in common: 272 of 273 or 99.63 %
1844 :Corresponding object --total pixel in common: 118 of 118 or 100.00 %
1595 :No Corresponding object --total pixel in common: 34 of 2005 or 1.70 %
***** Reference Object: 3 *****
1105 :No Corresponding object --total pixel in common: 13 of 130 or 10.00 %
768 :No Corresponding object --total pixel in common: 25 of 3020 or 0.83 %
713 :Corresponding object --total pixel in common: 3432 of 3437 or 99.85 %
and total pixel percent from reference to segment object 77.31 %
598 :No Corresponding object --total pixel in common: 4 of 657 or 0.61 %
1237 :No Corresponding object --total pixel in common: 5 of 137 or 3.65 %
1306 :No Corresponding object --total pixel in common: 9 of 340 or 2.65 %
1126 :Corresponding object --total pixel in common: 287 of 300 or 95.67 %
734 :Corresponding object --total pixel in common: 514 of 572 or 89.86 %
677 :Corresponding object --total pixel in common: 146 of 146 or 100.00 %
638 :No Corresponding object --total pixel in common: 2 of 301 or 0.66 %
729 :No Corresponding object --total pixel in common: 2 of 112 or 1.79 %

#print('object reference | common pixel | over-segment | under-segment |
union | number of corresponding segment objects | number of no corresponding
segment objects')

obref | copi | ovse | unse | un | numcorobj | numnocorobj
[[ 1. 1265. 35. 183. 1483. 6. 4.]
 [ 2. 504. 259. 66. 829. 3. 4.]
 [ 3. 4379. 60. 76. 4515. 4. 7.]]
-----

```

ΤΜΗΜΑ ΣΤ: Τελικός υπολογισμός των δεικτών για την υλοποίηση του ελέγχου ακρίβειας της κατάτμησης

```

sum_copi=0#overlapped
sum_ovse=0#over segment
sum_unse=0#under segment
sum_un=0#union
sum_ref=0#sum of reference
sum_seg=0#sum of segment
sum_cor_obj=0#sum of corresponding objects

for i in range (0,(int(refmax))):
    sum_copi=sum_copi+results_pixel[i][1]
    sum_ovse=sum_ovse+results_pixel[i][2]
    sum_unse=sum_unse+results_pixel[i][3]
    sum_un=sum_un+results_pixel[i][4]
    sum_ref=sum_ref+array_pixel_ref[i+1][1]
    sum_seg=sum_seg+results_pixel[i][1]+results_pixel[i][3]
    sum_cor_obj=sum_cor_obj+results_pixel[i][5]
print '-----'
print 'Total reference polygons:%.0f'%(int(refmax))

```

```

print 'Total corresponding segmentation objects:%.0f'%sum_cor_obj
print 'Total pixel in overlap:%.0f'%sum_copi
print 'Total pixel in oversegment:%.0f'%sum_ovse
print 'Total pixel in undersegment:%.0f'%sum_unse
print 'Total pixel in union:%.0f'%sum_un
print 'Total pixel of reference objects:%.0f'%sum_ref
print 'Total pixel of corresponding segmentation objects:%.0f'%sum_seg

print '-----'

QR=1-(sum_copi/sum_un)
OR=sum_ovse/sum_ref
UR=sum_unse/sum_seg
ED1=math.sqrt((OR*OR+UR*UR)/2)
PSE=sum_unse/sum_ref
NSR=abs(int(refmax)-sum_cor_obj)/int(refmax)
ED2=math.sqrt(PSE*PSE+NSR*NSR)

print 'QR=%.3f'%QR
print 'OR=%.3f'%OR
print 'UR=%.3f'%UR
print 'ED1=%.3f'%ED1
print 'PSE=%.3f'%PSE
print 'NSR=%.3f'%NSR
print 'ED2=%.3f'%ED2

```

Στο τελευταίο τμήμα του κώδικα γίνεται χρήση του πίνακα που δημιουργήθηκε στο E τμήμα και υπολογίζονται οι δείκτες για τον έλεγχο ακρίβειας της κατάτμησης. Ουσιαστικά αθροίζονται ανά στήλη οι γραμμές του πίνακα και τα προκύπτοντα αποτελέσματα χρησιμοποιούνται για τον υπολογισμό των δεικτών ED2 και ED1, ανάλογα με τις τιμές των οποίων λαμβάνει και ανάλογο χαρακτηρισμό η κατάτμηση που υλοποιήθηκε. Το τελικό αποτέλεσμα, που αποτελεί και το ζητούμενο στην ανάπτυξη του κώδικα αυτού, προκύπτει ως συνέχεια του τμήματος E και είναι:

```

Total reference polygons:3
Total corresponding segmentation objects:13
Total pixel in overlap:6148
Total pixel in oversegment:354
Total pixel in undersegment:325
Total pixel in union:6827
Total pixel of reference objects:6502
Total pixel of corresponding segmentation objects:6473

-----

QR=0.099
OR=0.054
UR=0.050

```

```
ED1=0.052
PSE=0.050
NSR=3.333
ED2=3.334
```

3.2.3 Ανάλυση του γενετικού αλγορίθμου

Στο εδάφιο αυτό αναλύεται το τελευταίο τμήμα του κώδικα μέσω του οποίου υλοποιείται ο ΓΑ. Ως συνάρτηση αξιολόγησης χρησιμοποιείται είτε ο δείκτης ED1, είτε ο δείκτης ED2 είτε ο συνδυασμός αυτών ως $(ED1+ED2)/2$. Ο ΓΑ αποτελείται από τα παρακάτω τμήματα κώδικα:

ΤΜΗΜΑ Α: Καθορισμός χαρακτηριστικών για την δημιουργία των χρωμοσωμάτων της αρχικής και των επόμενων γενιών

```
# -----Genome instance-----

genome = G1DList.G1DList(3)

genome.setParams(rangemin=1, rangemax=250) (A)

#setOfAlleles = GAllele.GAlleles(homogeneous=True)
##setOfAlleles = GAllele.GAlleles()
##a = GAllele.GAlleleRange(1, 150, real=False)
##setOfAlleles.add(a)
##a = GAllele.GAlleleRange(30, 250, real=False)
##setOfAlleles.add(a)
##a = GAllele.GAlleleRange(30,250, real=False)
##setOfAlleles.add(a)
#genome.setParams(allele=setOfAlleles) (B)

genome.initializator.set(Initializators.G1DListInitializatorInteger)

genome.crossover.set(Crossovers.G1DListCrossoverUniform)
genome.mutator.set(Mutators.G1DListMutatorIntegerGaussian)

genome.evaluator.set(evaluation)
```

Στο τμήμα αυτό αρχικά ορίζεται το πλήθος των μεταβλητών τις τιμές των οποίων καλείται να βελτιστοποιήσει ο ΓΑ. Εν συνεχεία προσδιορίζεται το εύρος τιμών για κάθε παράμετρο. Υπάρχει η δυνατότητα είτε να δοθεί κοινό εύρος τιμών για όλες τις παραμέτρους (A) είτε διαφορετικό για κάθε παράμετρο (B). Εν συνεχεία καθορίζεται ο τρόπος αρχικοποίησης των ατόμων της πρώτης γενιάς καθώς και οι τρόποι για την διασταύρωση και την μετάλλαξη που θα εφαρμοστούν στα παιδιά – γονίδια ώστε να προκύψουν οι επόμενες γενιές. Στο εδάφιο 2.3 αναπτύχθηκαν αναλυτικά οι δυνατοί τρόποι για την υλοποίηση των μεθόδων αυτών. Στο τέλος καθορίζεται η συνάρτηση αξιολόγησης με βάση την οποία αξιοποιεί ο αλγόριθμος το κάθε παιδί – γονίδιο (evaluation).

ΤΜΗΜΑ Β: Εισαγωγή παραμέτρων λειτουργίας του ΓΑ

```

#----- Genetic Algorithm Instance-----
ga=GSimpleGA.GSimpleGA(genome)
ga.selector.set(Selectors.GRouletteWheel)
ga.terminationCriteria.set(GSimpleGA.FitnessStatsCriteria)#statistika
ga.minimax = Consts.minimaxType["minimize"]

ga.setGenerations(10)
ga.setPopulationSize(100)
ga.setCrossoverRate(0.9)
ga.setMutationRate(0.02)

# -----set elitism-----
ga.setElitism(True)
ga.setElitismReplacement(2)

# -----evolve-----
ga.evolve(freq_stats=1)

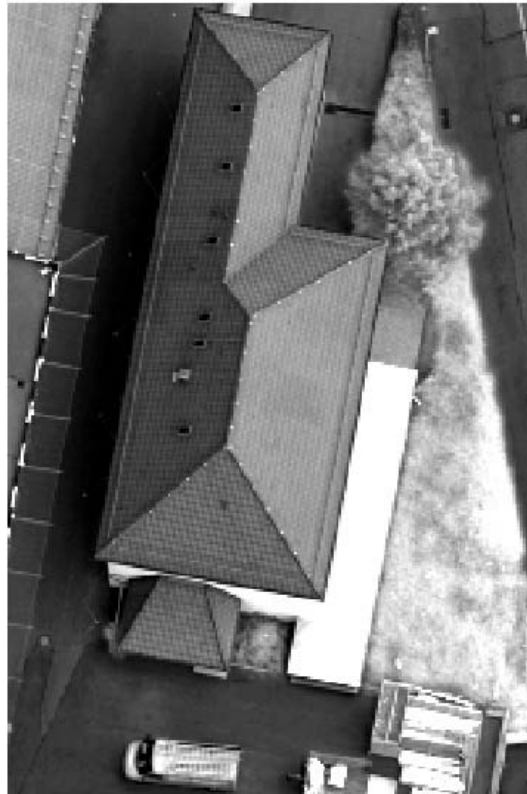
# -----Best individual-----
print ga.bestIndividual()

```

Στο δεύτερο και τελευταίο τμήμα του ΓΑ καθορίζονται τα χαρακτηριστικά του ΓΑ για την ορθή λειτουργία του. Συγκεκριμένα εισάγονται όλες οι επιλογές δημιουργίας των χρωμοσωμάτων (Τμήμα Α του κώδικα) ως παράμετροι του ΓΑ. Εν συνεχεία καθορίζεται ο τρόπος επιλογής των παιδιών στα οποία θα εφαρμοστούν οι μέθοδοι – επιλογές του ΓΑ δηλαδή η διασταύρωση και η μετάλλαξη. Επίσης καθορίζεται ένα δίνεται η δυνατότητα στον ΓΑ να συγκλίνει πριν την εξέταση όλων των γενιών, για στατιστικούς ή άλλους λόγους. Το επόμενο στάδιο το οποίο είναι και το πλέον σημαντικό και το οποίο μπορεί να οδηγήσει τον ΓΑ σε ορθά ή λανθασμένα αποτελέσματα σε περίπτωση λάθους επιλογής είναι ο καθορισμός του ελέγχου της τιμής της συνάρτησης αξιολόγησης ώστε ο ΓΑ να καταλαβαίνει ότι βελτιστοποιούνται οι τιμές των παραμέτρων. Αυτό επιτυγχάνεται με ελαχιστοποίηση της τιμής της συνάρτησης αξιολόγησης (minimize). Αν στην περίπτωση αυτή δοθεί μεγιστοποίηση της τιμής τότε τα αποτελέσματα θα είναι τα αντίθετα από τα αναμενόμενα. Ακολουθεί η υλοποίηση της λειτουργίας του ελιτισμού η οποία αναλύθηκε στο εδάφιο 2.3.7 και τέλος υλοποιείται η λειτουργία του ΓΑ (ga.evolve).

3.3 Χρησιμοποιηθείσες Εικόνες

Οι εικόνες που χρησιμοποιήθηκαν για την δοκιμή του αλγορίθμου προέρχονται από διάφορες πηγές. Συγκεκριμένα χρησιμοποιήθηκαν τέσσερις εικόνες. Η πρώτη προέρχεται από δεδομένα της Intergraph και αφορά εικόνα από την αερομεταφερόμενη κάμερα DMC, για την περιοχή calw της Γερμανίας, με μέγεθος εικονοστοιχείου 1 μέτρο. Η εικόνα αυτή χρησιμοποιήθηκε αρχικά ώστε να εκτελεστούν δοκιμές επί αυτής για την ορθή λειτουργία των αλγορίθμων (εικόνα 3.2). Η δεύτερη εικόνα προέρχεται από δεδομένα της εταιρίας Torosys και αφορά εικόνα από αερομεταφερόμενο σαρωτή Falcon II, για περιοχή της Γερμανίας με μέγεθος εικονοστοιχείου 1 μέτρο (εικόνα 3.3). Η τρίτη είναι εικόνα IKONOS για την περιοχή του Αγ. Στεφάνου Αττικής με μέγεθος εικονοστοιχείου 1 μέτρο (εικόνα 3.4). Τέλος η τέταρτη προέρχεται από την NASA, απεικονίζει τον πλανήτη Άρη και έχει μέγεθος εικονοστοιχείου 1 μέτρο (εικόνα 3.5).



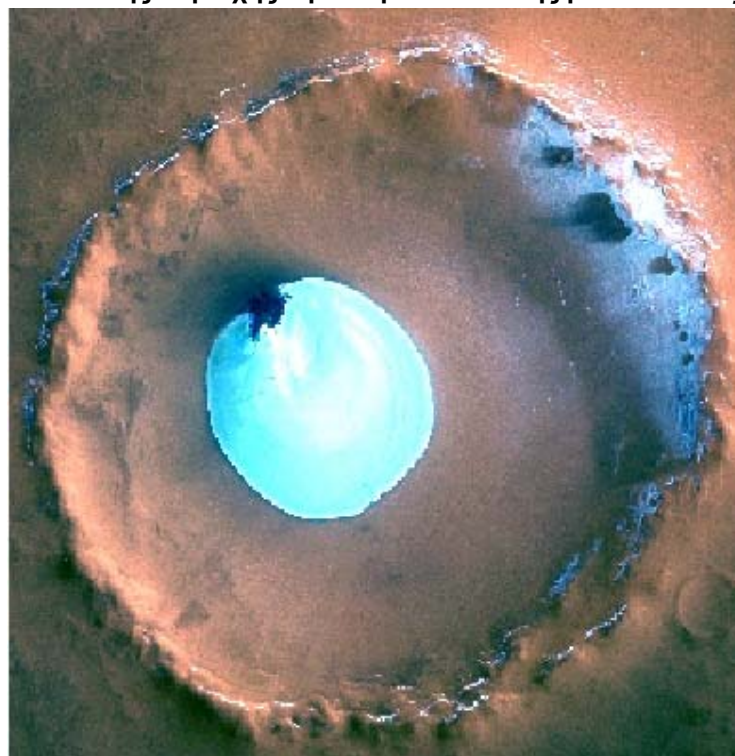
Εικόνα 3.2 - Εικόνα από αερομεταφερόμενη κάμερα DMC σε περιοχή της Γερμανίας με διαστάσεις 200 x 300 pixel



Εικόνα 3.3 - Εικόνα από αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας με διαστάσεις 473 x 474 pixel



Εικόνα 3.4 - ΔΕ Ικονος της περιοχής Αγ. Στεφάνου Αττικής με διαστάσεις 501 x 501 pixel

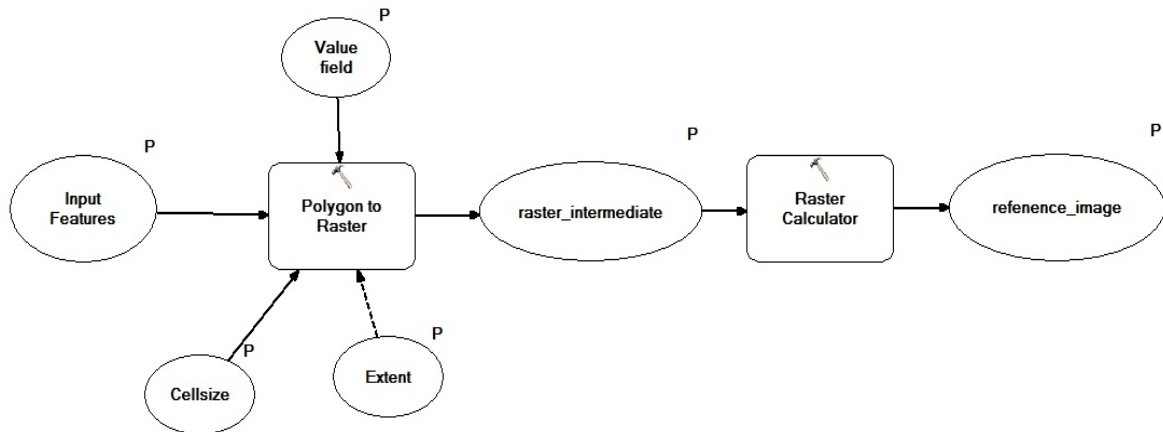


Εικόνα 3.5 - ΔΕ NASA του πλανήτη Άρη με διαστάσεις 299 x 307 pixel

3.4 Δημιουργία Εικόνων Αναφοράς

Όπως αναφέρθηκε ήδη στον υπολογισμό της συνάρτησης αξιολόγησης λαμβάνει μέρος η εικόνα αναφοράς, δηλαδή το βέλτιστο αποτέλεσμα στο οποίο καλείται να καταλήξει ο ΓΑ για συγκεκριμένα τμήματα της εικόνας. Η δημιουργία της εικόνας αυτής για κάθε εικόνα που χρησιμοποιήθηκε κατά την διαδικασία εφαρμογής του ΓΑ, πραγματοποιήθηκε μέσω του περιβάλλοντος του λογισμικού ArcMap10 με την χρήση του εργαλείου model builder.

Συγκεκριμένα δημιουργήθηκε το εργαλείο **Vect_to_Raster**. Η γραφική απεικόνιση αυτού, η οποία περιλαμβάνει τις λειτουργίες που υλοποιούνται ώστε να προκύψει το τελικό αποτέλεσμα παρουσιάζονται στην εικόνα 3.6.



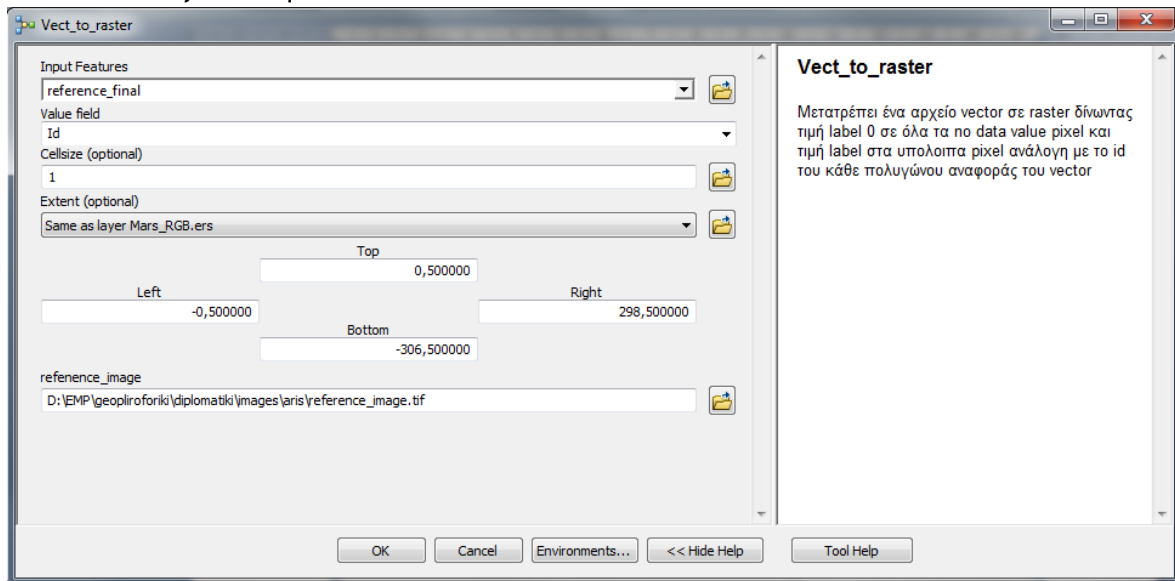
Εικόνα 3.6 - Γραφική απεικόνιση του εργαλείου Vect_to_Raster μέσω του Model Builder του λογισμικού ESRI ArcGIS 10

Προκειμένου να γίνει χρήση του εργαλείου αυτού πρέπει αρχικά να δημιουργηθεί ένα πολυγωνικό διανυσματικό αρχείο (shape file) στο οποίο να περιέχονται τα πολύγωνα αναφοράς. Σε κάθε ένα πολύγωνο αναφοράς δίνεται μια αρίθμηση με αύξουσα σειρά από το 1 έως το πλήθος των πολυγώνων. Η μορφή του πίνακα ιδιοτήτων του αρχείου αυτού στην τελική του μορφή απεικονίζεται στη εικόνα 3.7.

FID	Shape *	Id
0	Polygon	1
1	Polygon	2
2	Polygon	3
3	Polygon	4
4	Polygon	5
5	Polygon	6
6	Polygon	7
7	Polygon	8
8	Polygon	9
9	Polygon	10
10	Polygon	11
11	Polygon	12
12	Polygon	13

Εικόνα 3.7 - Μορφή πίνακα ιδιοτήτων πολυγωνικού αρχείου για την δημιουργία της τελικής εικόνας αναφοράς

Μετά την δημιουργία του αρχείου αυτού υλοποιείται η χρήση του εργαλείου **Vect_to_Raster**. Συγκεκριμένα με το άνοιγμα του εργαλείου εμφανίζεται το παράθυρο που απεικονίζεται στην εικόνα 3.8.

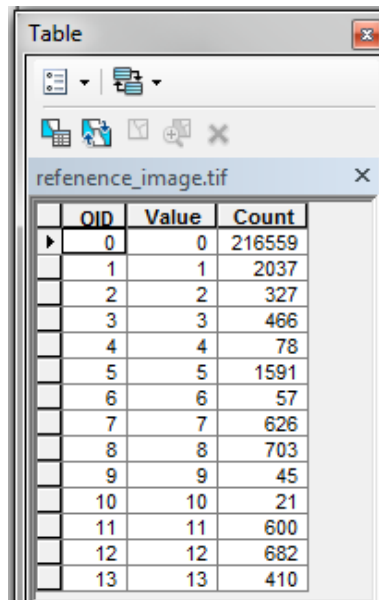


Εικόνα 3.8 - Εμφάνιση παραθύρου από την εφαρμογή του εργαλείου Vect_to_Raster

Στο παράθυρο αυτό υλοποιούνται οι παρακάτω επιλογές:

- **Input Features:** Εισαγωγή του διανυσματικού αρχείου που δημιουργήθηκε,
- **Value field:** Το πεδίο του πίνακα ιδιοτήτων όπου υπάρχει η αρίθμηση των πολυγώνων αναφοράς (Id),
- **Cellsize:** Η διάσταση του ρixel, η οποία πρέπει να είναι ίδια με την διάσταση του ρixel της αρχικής εικόνας, ώστε η εικόνα αναφοράς να έχει ίδιο μέγεθος εικονοστοιχείου αλλά και ίδιο πλήθος εικονοστοιχείων με την αρχική εικόνα,
- **Extend:** Η έκταση της εικόνας που θα δημιουργηθεί, της οποίας τα όρια πρέπει να είναι ίδια με τα όρια της αρχικής εικόνας και
- **Refenece_image:** Η διαδρομή στην οποία θα αποθηκευτεί το αποτέλεσμα του εργαλείου.

Ουσιαστικά μέσω του εργαλείου αυτού δημιουργείται μια εικόνα η οποία έχει τιμή εικονοστοιχείου 0 σε όλα αυτά που δεν περιέχονται σε κάποιο πολύγωνο αναφοράς και τιμή εικονοστοιχείου ίση με την τιμή αρίθμησης του κάθε πολυγώνου για αυτά που ανήκουν στο πολύγωνο. Ο πίνακας ιδιοτήτων του αποτελέσματος από την χρήση του εργαλείου στο διανυσματικό αρχείο που αναφέρθηκε εμφανίζεται στην εικόνα 3.9. Παρατηρείται ότι εμφανίζεται το πλήθος όλων των ρixel ανά τιμή του πολυγώνου αναφοράς καθώς και το πλήθος των ρixel με τιμή 0, δηλαδή όσα δεν ανήκουν στα πολύγωνα αναφοράς.

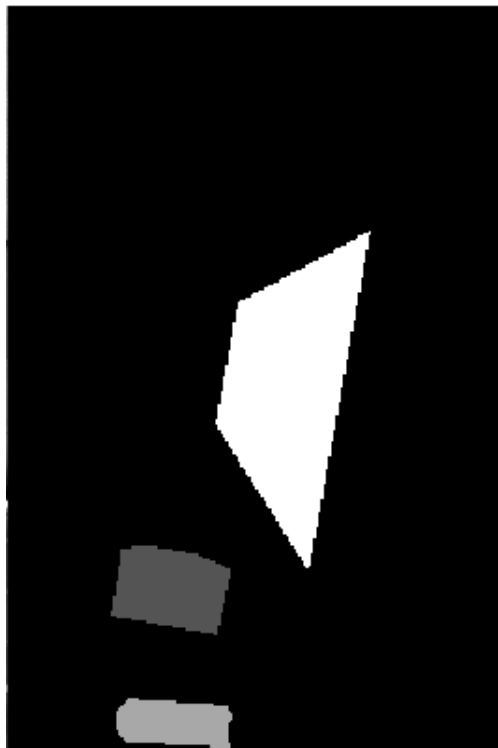


The screenshot shows a window titled 'Table' with a toolbar and a table of data. The table has three columns: 'OID', 'Value', and 'Count'. The data is as follows:

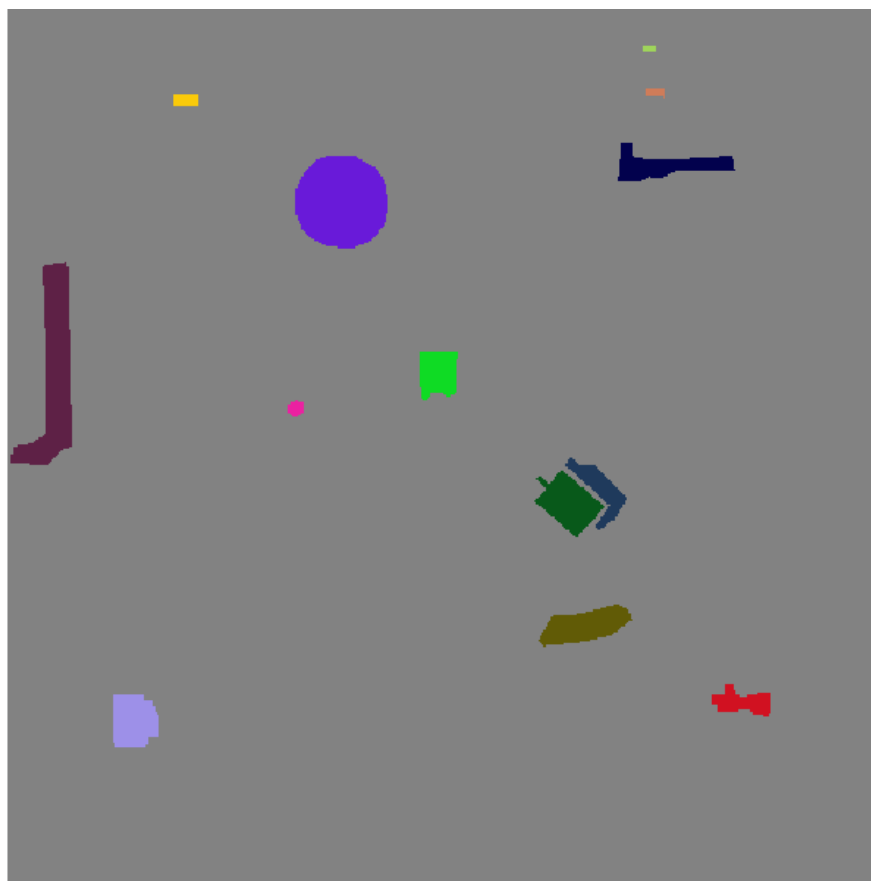
OID	Value	Count
0	0	216559
1	1	2037
2	2	327
3	3	466
4	4	78
5	5	1591
6	6	57
7	7	626
8	8	703
9	9	45
10	10	21
11	11	600
12	12	682
13	13	410

Εικόνα 3.9 - Μορφή πίνακα ιδιοτήτων της τελικής εικόνας αναφοράς

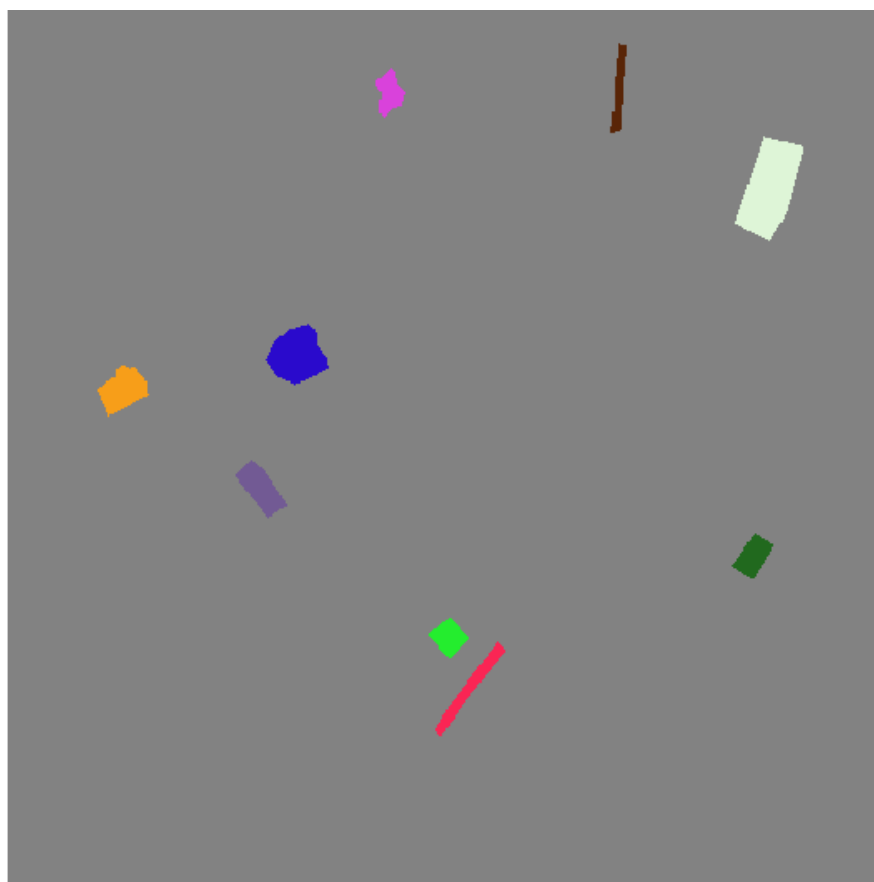
Οι εικόνες που προέκυψαν μετά και το τελευταίο βήμα είναι οι εξής:



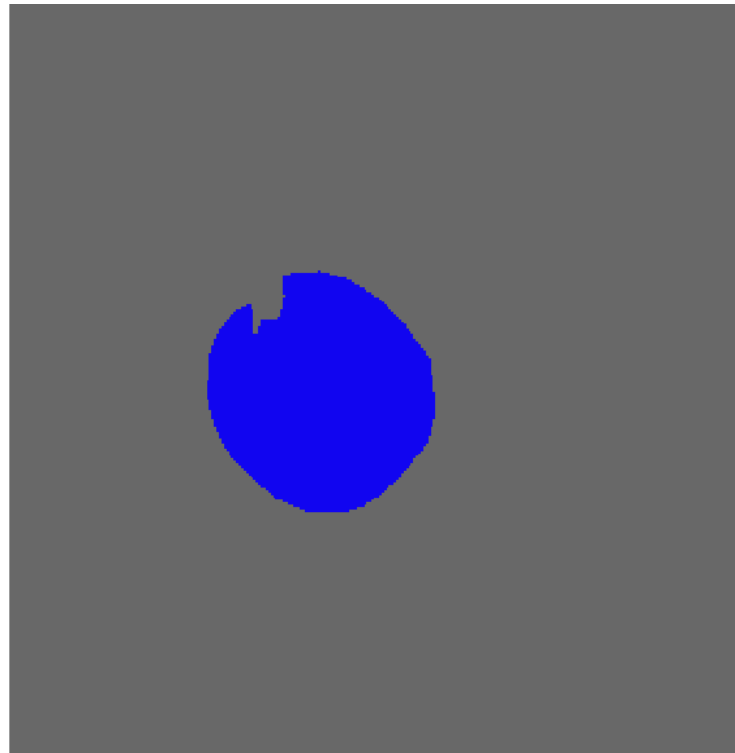
Εικόνα 3.10 - Εικόνα αναφοράς από την εφαρμογή του εργαλείου Vect_to_Raster στην εικόνα 3.2 (Περιοχή calw Γερμανίας, εταιρία Intergraph)



Εικόνα 3.11 - Εικόνα αναφοράς από την εφαρμογή του εργαλείου Vect_to_Raster στην εικόνα 3.3 (Περιοχή Γερμανίας εταιρία Toposys)



Εικόνα 3.12 - Εικόνα αναφοράς από την εφαρμογή του εργαλείου Vect_to_Raster στην εικόνα 3.4 (Περιοχή Αγ. Στεφάνου Αττικής)



Εικόνα 3.13 - Εικόνα αναφοράς από την εφαρμογή του εργαλείου Vect_to_Raster στην εικόνα 3.5 (Πλανήτης Άρης από δορυφόρο της NASA)

3.5 Ανάλυση Ευαισθησίας – Επιλογή Παραμέτρων

3.5.1 Γενικά

Στο κεφάλαιο αυτό παρουσιάζονται όλες οι δοκιμές που έγιναν στις εικόνες που αναφέρθηκαν, με την εφαρμογή του ΓΑ. Συγκεκριμένα:

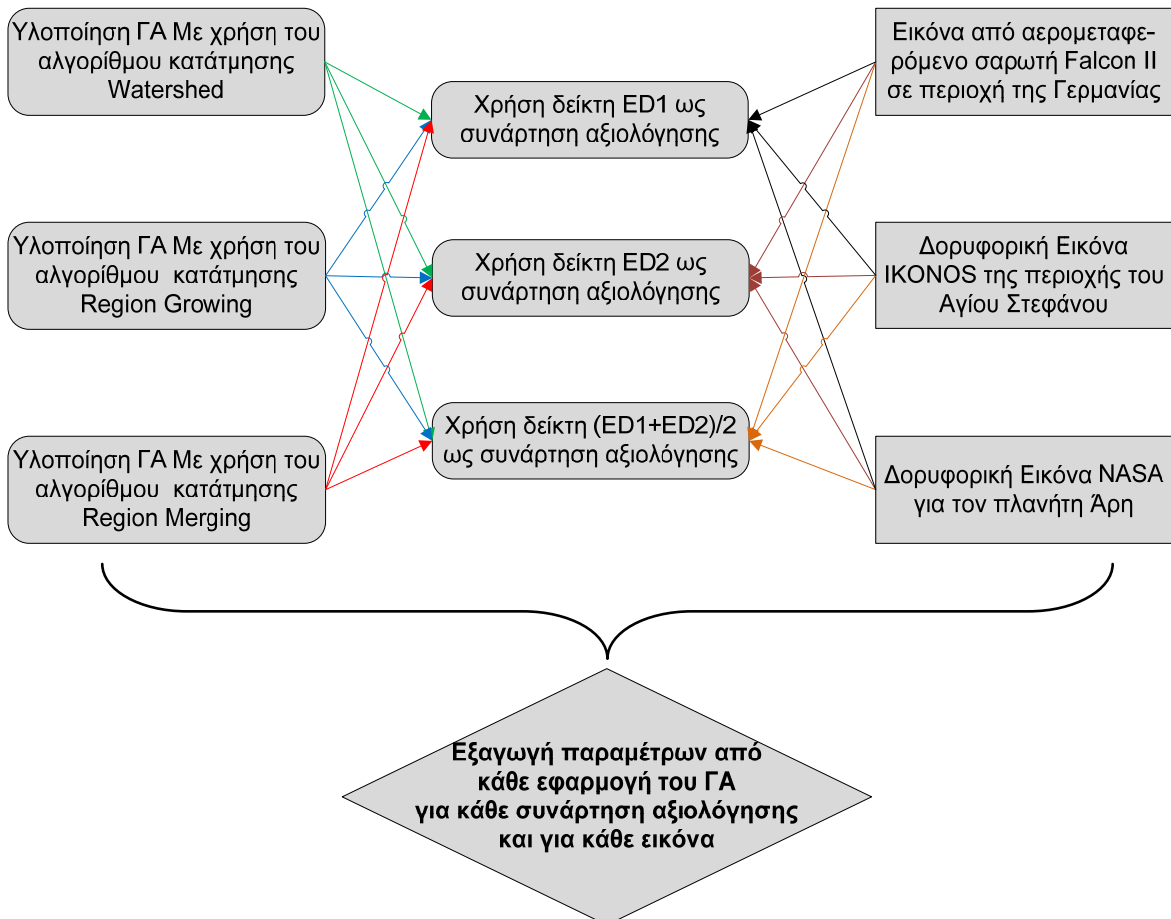
- Στο εδάφιο 3.5.2 παρουσιάζονται οι δοκιμές και τα αποτελέσματα, ανά εικόνα, χρησιμοποιώντας τον αλγόριθμο κατάτμησης που βασίζεται στον μετασχηματισμό του υδροκρίτη,
- Στο εδάφιο 3.5.3 παρουσιάζονται οι δοκιμές και τα αποτελέσματα, ανά εικόνα, χρησιμοποιώντας τον αλγόριθμο κατάτμησης που βασίζεται σε τεχνικές τοπικής ανάλυσης επέκτασης περιοχών και
- Στο εδάφιο 3.5.4 παρουσιάζονται οι δοκιμές και τα αποτελέσματα, ανά εικόνα, χρησιμοποιώντας τον αλγόριθμο κατάτμησης που βασίζεται σε τεχνικές τοπική ένωσης περιοχών.

Η ανάπτυξη των περιπτώσεων παρουσιάζεται με την εξής δομή:

- Παρουσίαση σε πίνακα, ανά εικόνα, των **εφαρμογών** του ΓΑ. Για κάθε εφαρμογή εμφανίζονται όλα τα στοιχεία που χρησιμοποιήθηκαν, ο δείκτης που χρησιμοποιήθηκε ως στοιχείο αξιολόγησης του ΓΑ καθώς και τα αποτελέσματα που προέκυψαν. Στους πίνακες αυτούς εμφανίζονται οι εξής στήλες.
 - ✓ **Ημερ. Έναρξης:** Ημερομηνία έναρξης της εφαρμογής του ΓΑ στην εκάστοτε εικόνα,
 - ✓ **Χρόνος Εκτ/σης:** Συνολικός χρόνος εκτέλεσης του ΓΑ στην εκάστοτε εικόνα,
 - ✓ **Pop. Size:** Το μέγεθος του πληθυσμού σε κάθε γενιά,
 - ✓ **Initialization:** Συνάρτηση δημιουργίας του αρχικού πληθυσμού,
 - ✓ **Selection:** Συνάρτηση επιλογής ατόμων για την δημιουργία της επόμενης γενιάς,
 - ✓ **Crossover:** Συνάρτηση διασταύρωσης επιλεγέντων ατόμων,
 - ✓ **Mutation:** Συνάρτηση μετάλλαξης ατόμων,
 - ✓ **Gen:** Σύνολο γενιών εφαρμογής του ΓΑ,
 - ✓ **Mutation rate:** Ρυθμός μετάλλαξης ατόμων,
 - ✓ **Crossover rate:** Ρυθμός διασταύρωσης ατόμων,
 - ✓ **LB:** Κατώτερα όρια τιμών των εκάστοτε παραμέτρων του ΓΑ,
 - ✓ **UB:** Ανώτερα όρια τιμών των παραμέτρων του ΓΑ,
 - ✓ **Συν.Αξιολ.:** Συνάρτηση αξιολόγησης του ΓΑ όπου αναφέρεται είτε ED2, είτε ED1 είτε $(ED1+ED2)/2$ και
 - ✓ **ΑΠΟΤΕΛΕΣΜΑΤΑ:** Παρουσιάζονται οι τιμές των παραμέτρων από την εφαρμογή του ΓΑ, η τιμή του δείκτη ED1 ή ED2 ή $(ED1+ED2)/2$ σαν αρχική τιμή και σαν τιμή αξιολόγησης.

- Εμφάνιση των **κατατμήσεων** που προέκυψαν με βάση τα αποτελέσματα εφαρμογής του ΓΑ για κάθε εικόνα. Προκειμένου να υπάρχει οπτική σύγκριση παρουσιάζεται η εικόνα κατάτμησης μαζί με την εικόνα αναφοράς καθώς και τα επιμέρους τμήματα όπου εμφανίζονται τα αντικείμενα αναφοράς.

Το διάγραμμα ροής για την διαδικασία που θα αναλυθεί έχει ως εξής:



Εικόνα 3.14 - Διάγραμμα ροής εφαρμογής του ΓΑ με βάση τους αλγορίθμους κατάτμησης και τις χρησιμοποιηθέντες εικόνες

3.5.2 Υλοποίηση Γενετικού Αλγορίθμου, με Χρησιμοποίηση του Αλγορίθμου Κατάτμησης Watershed

Στην ενότητα αυτή πραγματοποιείται εφαρμογή του ΓΑ με χρησιμοποίηση του αλγορίθμου κατάτμησης Watershed. Αρχικά υλοποιήθηκαν πλήθος από δοκιμές στην εικόνα 3.2 (Calw) που ωστόσο λόγω του πειραματικού τους χαρακτήρα αλλά και των μη αναμενόμενων, πολλές φορές, αποτελεσμάτων λόγω διερεύνησης της διαδικασίας δεν αναφέρονται στα εδάφια που ακολουθούν. Στα εδάφια 3.5.2.1, 3.5.2.2 και 3.5.2.3 παρουσιάζονται τα αποτελέσματα για κάθε μια από τις τρεις εικόνες που χρησιμοποιούνται στην διαδικασία υλοποίησης του ΓΑ.

3.5.2.1 Υλοποίηση στην ΔΕ του Αγίου Στεφάνου Αττικής

Τα αποτελέσματα των κατατμήσεων που προέκυψαν χρησιμοποιώντας τον συγκεκριμένο αλγόριθμο κατάτμησης στην υλοποίηση του ΓΑ, εξαρτώνται από την συνάρτηση αξιολόγησης που χρησιμοποιείται κάθε φορά. Καλύτερα αποτελέσματα παρατηρούνται με την χρησιμοποίηση του δείκτη ED2, σε σχέση με τους υπόλοιπους δυο, ως προς το αποτέλεσμα της κατάτμησης.

Όσον αφορά τον χρόνο εκτέλεσης του ΓΑ αυτός κυμαίνεται από 18,5 ώρες έως 51,8 ώρες ανάλογα με τις επιλογές του ΓΑ και κυρίως με το πλήθος γενεών και τον εκάστοτε πληθυσμό. Ωστόσο όπως παρατηρείται και στον πίνακα 3.1 μεγαλύτερος χρόνος εφαρμογής του αλγορίθμου δεν συνεπάγεται και καλύτερα αποτελέσματα. Για παράδειγμα για την συνάρτηση αξιολόγησης ED2 καλύτερα αποτελέσματα προέκυψαν σε 18,5 ώρες στην δεύτερη περίπτωση του πίνακα 3.1, όπου είναι και ο μικρότερος χρόνος εφαρμογής του αλγορίθμου. Η διαφορά μέγιστου και ελάχιστου χρόνου οφείλεται στις διαφορετικές διαδικασίες υλοποίησης του ΓΑ σε κάθε εφαρμογή του. Ιδίως το πλήθος του πληθυσμού και των γενεών είναι οι καθοριστικοί παράγοντες του χρόνου εκτέλεσης του ΓΑ. Για τον λόγο αυτό πρέπει αυτοί να επιλέγονται με ιδιαίτερη προσοχή ώστε να επιτυγχάνονται τα βέλτιστα αποτελέσματα στον συντομότερο χρόνο. Το πεδίο έρευνας του ΓΑ δεν πρέπει ούτε να περιορίζεται αλλά ούτε και να είναι εντελώς διευρυμένο γιατί στην μεν πρώτη περίπτωση τα αποτελέσματα του ΓΑ εμπριέχουν πολύ μεγάλο κίνδυνο να είναι εσφαλμένα ενώ στην δεύτερη περίπτωση θα δώσει κατά μεγάλη πιθανότητα ορθά αποτελέσματα αλλά σε πολύ μεγάλο χρόνο, γεγονός που μπορεί να καταστήσει τον ΓΑ μη συμφέρων.

Όσον αφορά τις επιλογές του ΓΑ στις διαδικασίες υλοποίησης του, επιλέχτηκαν σε κάθε εφαρμογή όσο το δυνατόν διαφορετικές, ώστε να ελεγχθούν τα αποτελέσματα που προκύπτουν. Ωστόσο λόγω του ότι το εύρος τιμών των παραμέτρων δεν ήταν το ίδιο για όλες, δημιούργησε περιορισμό στον τρόπο αρχικοποίησης του αρχικού πληθυσμού καθώς και στον τρόπο διασταύρωσης και μετάλλαξης, καθώς κάποιες μέθοδοι δεν υποστήριζαν την συγκεκριμένη υλοποίηση. Για το λόγο αυτό η αρχικοποίηση και η μετάλλαξη υλοποιήθηκε σε όλες τις εφαρμογές του ΓΑ με την επιλογή «Allele». Ωστόσο οι υπόλοιπες επιλογές του ΓΑ τροποποιούνταν σε κάθε εφαρμογή. Για παράδειγμα ο ρυθμός διασταύρωσης και μετάλλαξης καθώς και το εύρος τιμών των παραμέτρων, όπως φαίνεται και στον πίνακα 3.1.

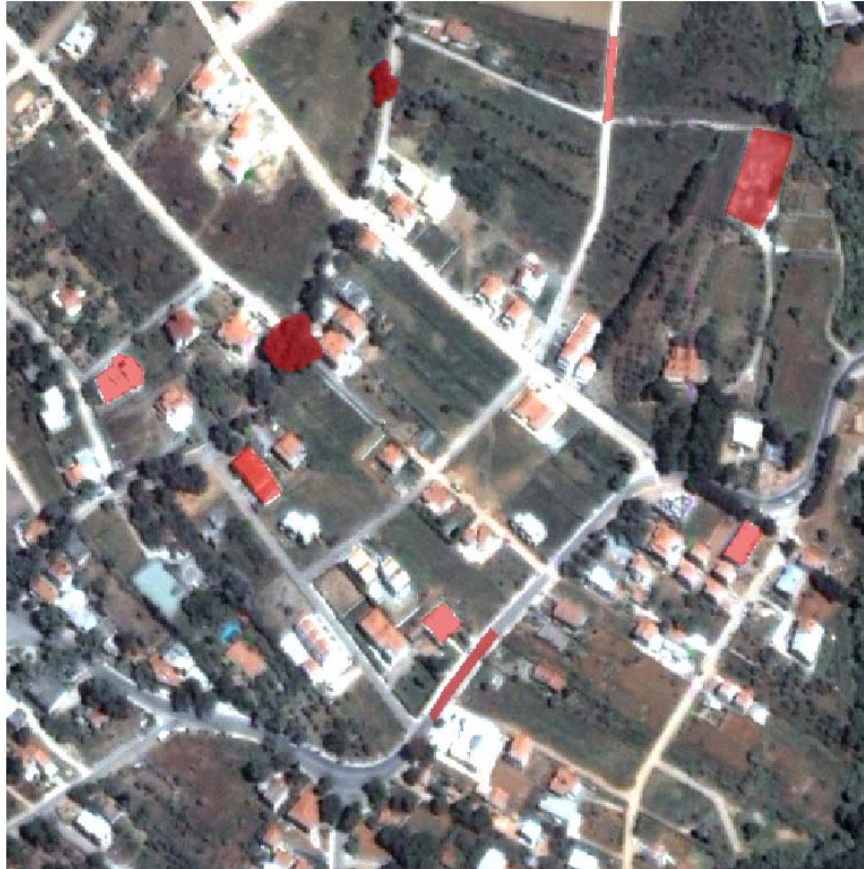
Για στατιστικούς λόγους αναφέρεται ότι ο συνολικός χρόνος εκτέλεσης των οκτώ εφαρμογών ανέρχεται σε περίπου 248 ώρες με μέσο όρο τις 31 ώρες.

ΔΕ Άγιος Στέφανος 501pixel x 501pixel

Ημερ Έναρξης	Χρόνος Εκτ/σης	Pop Size	Initiali- zation	Selection	Crossover	Mutation	Gen	mutation rate	crossover rate	LB (P1_P2_P3), P4	UB (P1_P2_P3) , P4	Συνάρτηση Αξιολόγησης	ΑΠΟΤΕΛΕΣΜΑΤΑ				
													P1	P2	P3	P4	Min Fitness /Min Raw
10/2/13	51,8h	60	Allele (integer)	Tourna- ment	TwoPoint	Allele	20	3%	90%	0_0_0 3 ή 5 ή 7	255_255_10 3 ή 5 ή 7	ED2	114	247	3	3	9,45/ 0,71
11/2/13	18,5h	40	Allele (real)	Rank	TwoPoint	Allele	20	3%	95%	1_1_1 3 ή 5 ή 7	255_255_9 3 ή 5 ή 7	ED2	195	229	3	5	0,82/ 0,70
13/2/13	24,9h	80	Allele (integer)	Roulette Wheel	Uniform	Allele	10	2%	100%	1_1_1 3 ή 5 ή 7	255_255_9 3 ή 5 ή 7	ED2	220	243	3	3	2,19/ 0,71
15/2/13	30,2h	80	Allele (integer)	Roulette Wheel	Uniform	Allele	10	2%	100%	1_1_0 3 ή 5 ή 7	255_255_9 3 ή 5 ή 7	ED1	22	179	0	5	0,28/ 0,11
17/2/13	33,2h	60	Allele (integer)	Tourna- ment	Uniform	Allele	20	3%	95%	1_1_1 3 ή 5 ή 7	255_255_15 3 ή 5 ή 7	ED1	41	239	1	3	0,29/ 0,25
18/2/13	24,2h	60	Allele (integer)	Tourna- ment	Uniform	Allele	15	3%	85%	1_1_1 3 ή 5 ή 7 ή 9	255_255_15 3 ή 5 ή 7 ή 9	ED1	104	189	1	5	0,23/ 0,26
19/2/13	21,8h	60	Allele (integer)	Tourna- ment	Uniform	Allele	15	3%	85%	1_1_1 3 ή 5 ή 7 ή 9	255_255_15 3 ή 5 ή 7 ή 9	(ED1+ED2)/2	192	40	3	3	1,49/0,61
22/2/13	44,2h	80	Allele (integer)	Roulette Wheel	Uniform	Allele	20	2%	80%	1_1_1 3 ή 5 ή 7 ή 9	255_255_15 3 ή 5 ή 7 ή 9	(ED1+ED2)/2	8	247	3	3	1,14/ 0,60

Πίνακας 3.1- Στοιχεία εφαρμογής ΓΑ στην ΔΕ ΙΚΟΝΟΣ περιοχής του Αγίου Στεφάνου Αττικής, με αλγόριθμο κατάτμησης watershed

Οι βέλτιστες κατατμήσεις που προκύπτουν για την εικόνα, για κάθε μια από τις τρεις συνάρτησης αξιολόγησης, με βάση τα αποτελέσματα του πίνακα 3.1 παρουσιάζονται στις εικόνες παρακάτω. Στην εικόνα 3.15 παρουσιάζονται τα αντικείμενα αναφοράς τα οποία επικαλύπτουν την ΔΕ του Αγίου Στεφάνου, για την εμποπτική παρουσίαση αυτών.

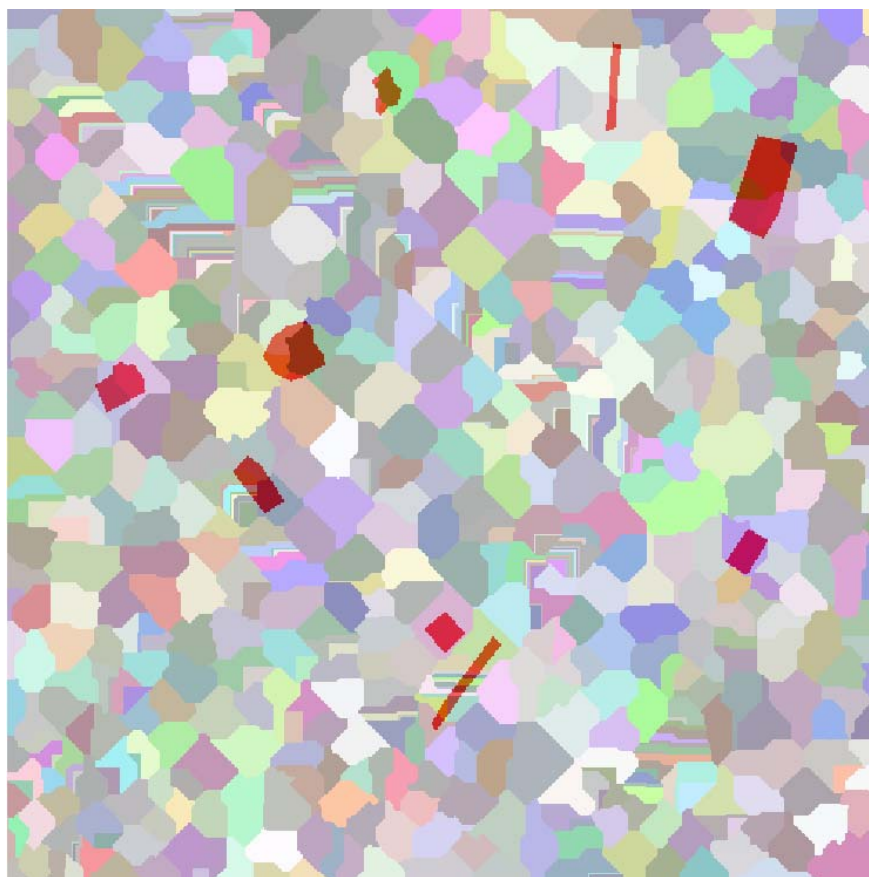


Εικόνα 3.15 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα του Αγ. Στεφάνου

Συνάρτηση αξιολόγησης ED2:

Η χρησιμοποίηση του δείκτη ED2 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED2=0,70$ με τιμές παραμέτρων $P1=195$, $P2=229$, $P3=3$ και $P4=5$. Για τις τιμές αυτές ο δείκτης ED1 λαμβάνει τιμή $ED1=0,488$. Παρατηρείται ότι η τιμή αυτή δεν είναι η βέλτιστη με την χρησιμοποίηση του ED1 ως δείκτη για την συνάρτηση αξιολόγησης, καθώς η βέλτιστη τιμή είναι η $ED1=0,11$. Το γεγονός αυτό αποδεικνύει ότι οι δυο δείκτες είναι διαφορετικοί μεταξύ τους και επιπλέον ότι ελαχιστοποίηση του ενός δεν συνεπάγεται ταυτόχρονη ελαχιστοποίηση και του άλλου δείκτη.

Από την κατάτμηση δημιουργούνται συνολικά 726 αντικείμενα. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση δεν μπορεί να χαρακτηριστεί ως η βέλτιστη δυνατή ωστόσο η χρήση του ΓΑ απέδωσε ικανοποιητικά αποτελέσματα. Γενικότερα η κατάτμηση με χρήση του αλγορίθμου κατάτμησης με την μέθοδο του υδροκρίτη δεν χαρακτηρίζεται ως βέλτιστη. Στις εικόνες 3.16 ως 3.22 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με τα αντικείμενα αναφοράς. Στην εικόνα 3.16 παρουσιάζεται το σύνολο της εικόνας ενώ στις υπόλοιπες τμηματικά τα αντικείμενα αναφοράς για καλύτερη εμποπτική παρουσίαση των αποτελεσμάτων σε σχέση με τα αντικείμενα αναφοράς.



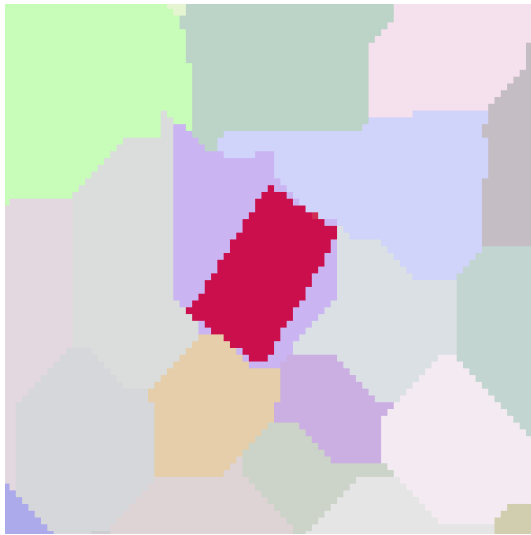
Εικόνα 3.16 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED2 (watershed - Min Fitness / Min Raw = 0,82 / 0,70)



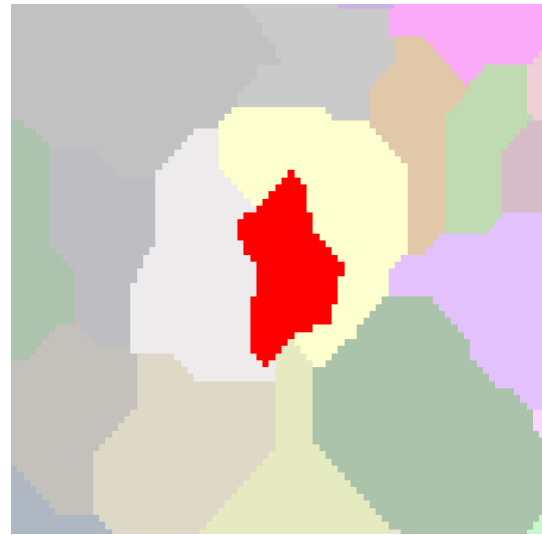
Εικόνα 3.17 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



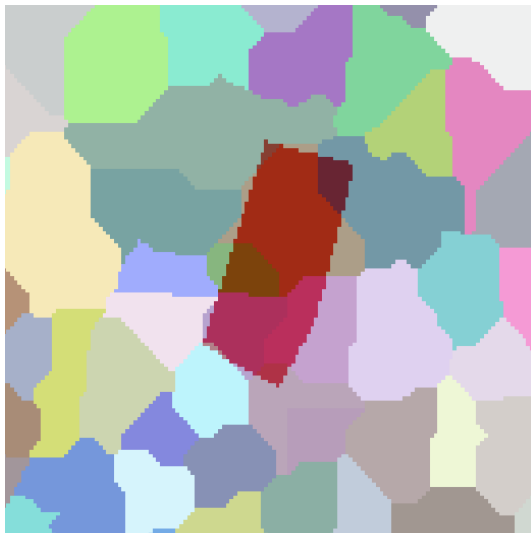
Εικόνα 3.18 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.19 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.20 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.21 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

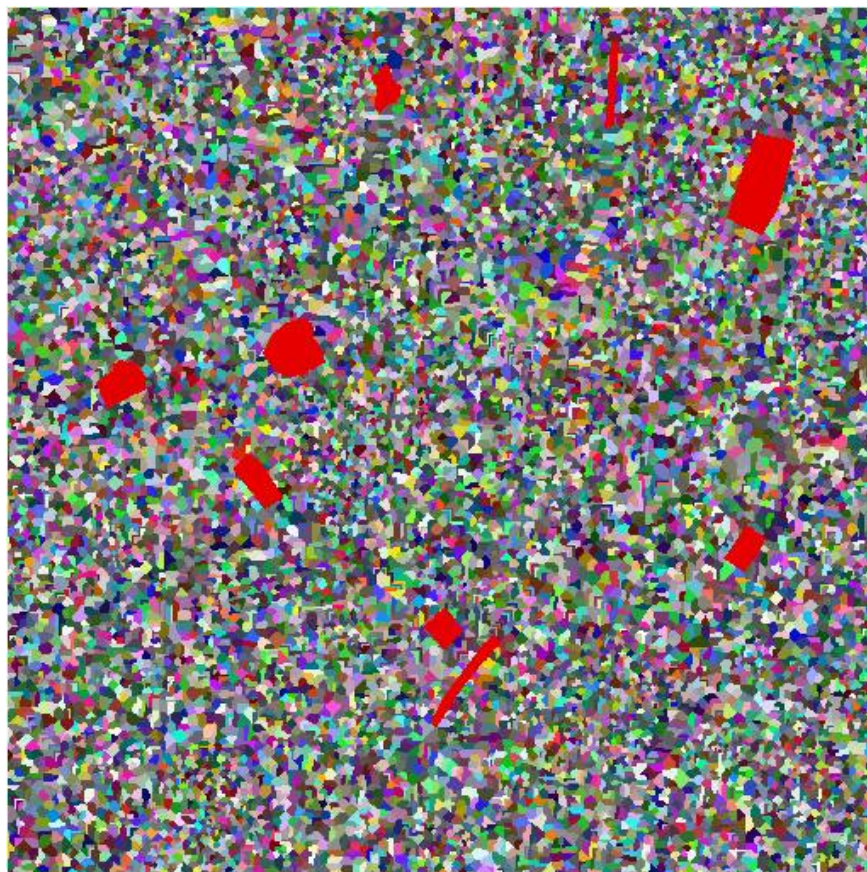


Εικόνα 3.22 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

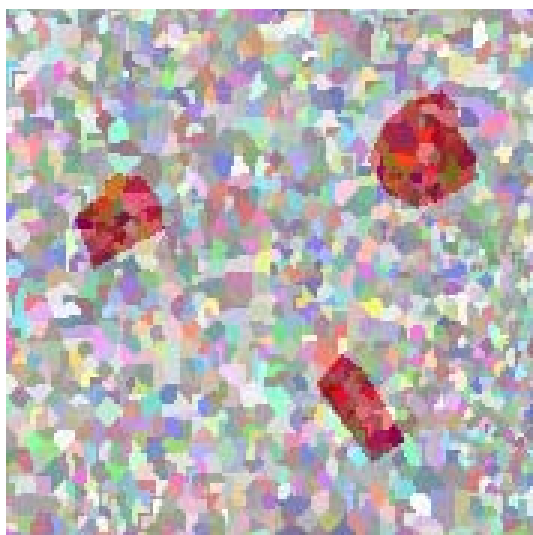
Συνάρτηση αξιολόγησης ED1:

Η χρησιμοποίηση του δείκτη ED1 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED1=0,11$ με τιμές παραμέτρων $P1=22$, $P2=179$, $P3=0$ και $P4=5$. Για τις τιμές αυτές ο δείκτης ED2 λαμβάνει τιμή $ED=24,667$. Το γεγονός αυτό ενδυναμώνει το συμπέρασμα ότι οι δυο δείκτες είναι διαφορετικοί και ότι ο καθένας από αυτούς συμπεριφέρεται με διαφορετικό τρόπο. Παρατηρείται ότι η τιμή της παραμέτρου P3 γίνεται μηδέν γεγονός που μειώνει την ορθή χρησιμοποίηση του δείκτη αυτού για την συνάρτηση αξιολόγησης.

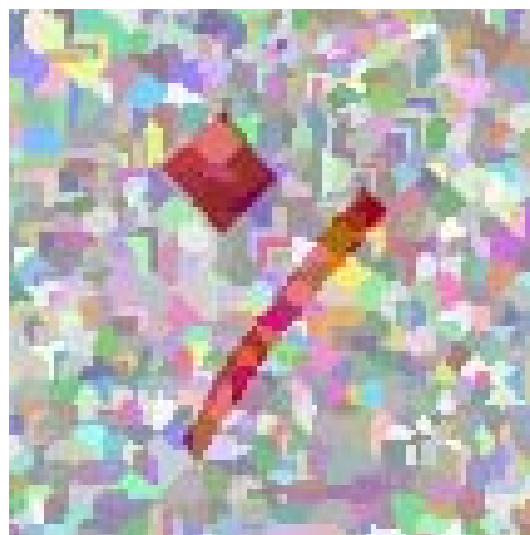
Από την κατάτμηση δημιουργούνται συνολικά 15599 αντικείμενα. Το μέγεθος αυτό είναι ιδιαίτερα μεγάλο και όπως παρουσιάζεται και στις εικόνες 3.23 έως 3.25 η κατάτμηση δεν μπορεί να χαρακτηριστεί ως ορθή. Τα αντικείμενα αναφοράς περιέχονται σε πάρα πολλά αντικείμενα κατάτμησης. Ο ΓΑ στην περίπτωση αυτή να μην λειτούργησε σωστά ώστε να ελαχιστοποιήσει τον δείκτη ED1, όμως ο δείκτης δεν αποτελεί ορθό μέτρο για την χρησιμοποίηση στην συνάρτηση αξιολόγησης, με αποτέλεσμα να λαμβάνονται τα παρακάτω αποτελέσματα.



Εικόνα 3.23 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED1 (watershed - Min Fitness / Min Raw = 0,28 / 0,11)

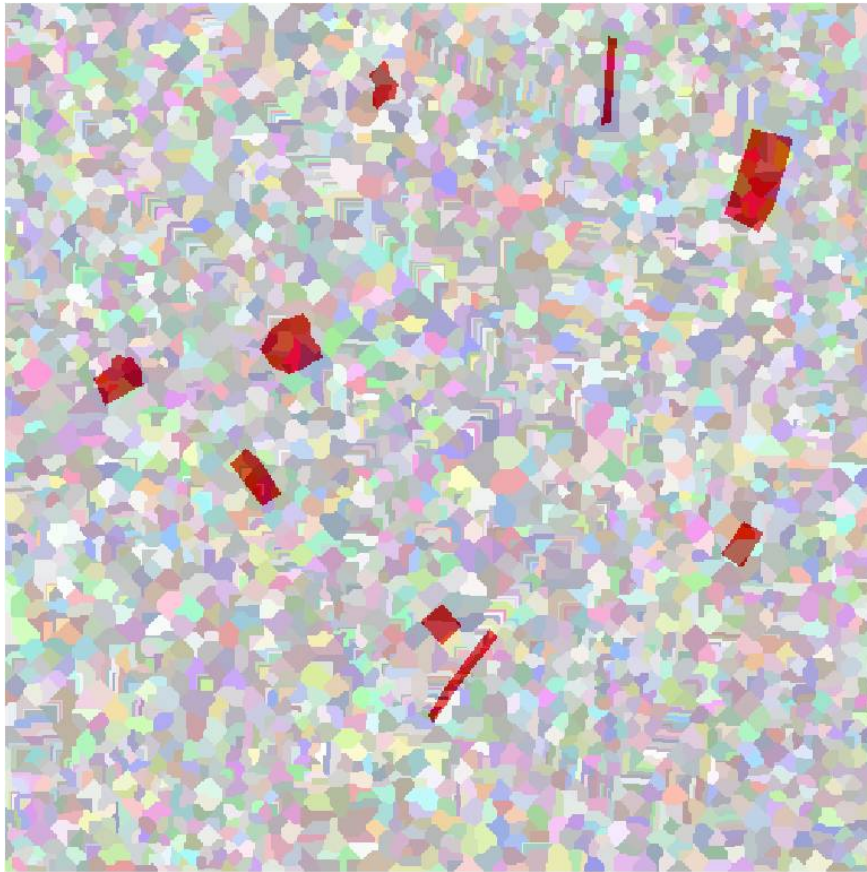


Εικόνα 3.24 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



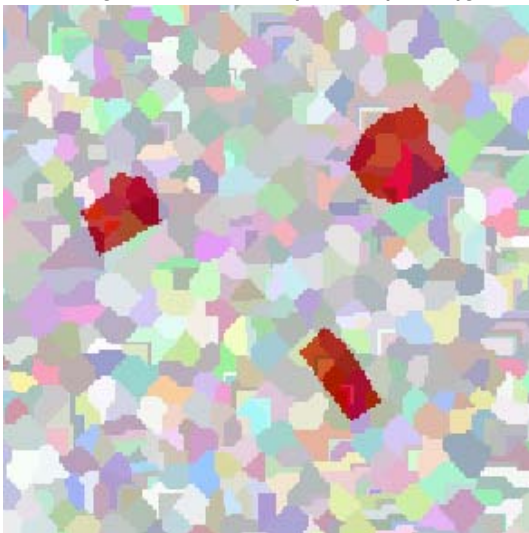
Εικόνα 3.25 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

Ωστόσο προκειμένου να παρουσιαστεί και η επόμενη κατάτμηση ανάλογα με την τιμή καταλληλότητας, εξαιτίας της μηδενικής τιμής στην τρίτη παράμετρο, παρουσιάζεται στην εικόνα 3.26, το αποτέλεσμα της εφαρμογής με $\text{min fitness}/\text{min raw} = 0,23 / 0,26$.



Εικόνα 3.26 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED1 (watershed - Min Fitness / Min Raw = 0,23 / 0,26)

Στην περίπτωση αυτή δημιουργούνται 3866 αντικείμενα, σαφώς λιγότερα σε σχέση με την προηγούμενη, ωστόσο και πάλι η κατάτμηση δεν μπορεί να χαρακτηριστεί ως βέλτιστη καθώς τα αντικείμενα αναφοράς δεν σχετίζονται με τα αντικείμενα κατάτμησης. Ο δείκτης ED2 λαμβάνει τιμή 4,559 μεγαλύτερη από την βέλτιστη και πάλι. Η δυνατότητα που δόθηκε στον ΓΑ να επιλέξει και την τιμή 0 οδήγησε μεν σε καλύτερη τιμή της συνάρτησης αξιολόγησης αλλά όχι σε καλύτερη κατάτμηση όπου δημιουργήθηκαν πάρα πολλά και μικρά αντικείμενα. Στις εικόνες 3.27 και 3.28 παρουσιάζεται ένα μέρος από τα αντικείμενα αναφοράς σε επικάλυψη με τμήμα της κατάτμησης.



Εικόνα 3.27 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

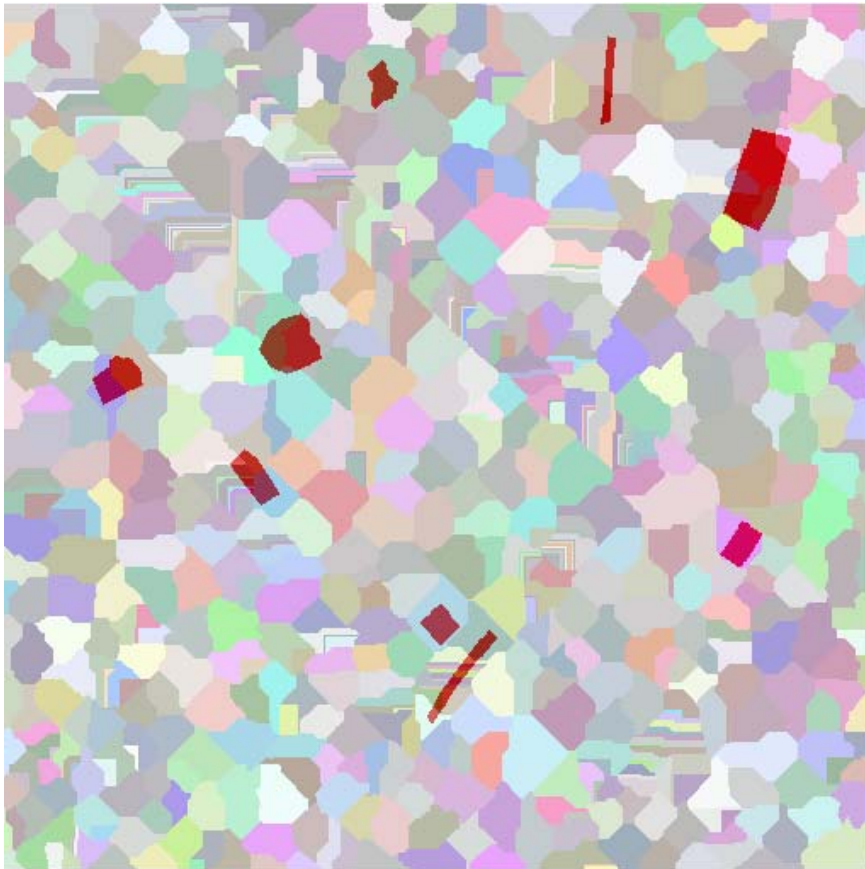


Εικόνα 3.28 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

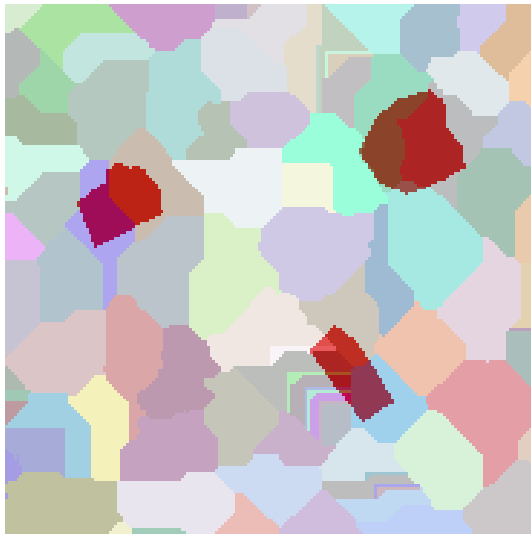
Συνάρτηση αξιολόγησης $(ED1+ED2)/2$:

Η χρησιμοποίηση του δείκτη $(ED1+ED)/2$ ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή 0,60. Συγκεκριμένα τιμές $ED1=0,486$ και $ED2=0,714$ με τιμές παραμέτρων $P1=8$, $P2=247$, $P3=3$ και $P4=3$. Παρατηρείται ότι με την χρησιμοποίηση του δείκτη αυτού η προκύπτουσα κατάτμηση ακολουθεί κυρίως τον δείκτη $ED2$. Ουσιαστικά βελτιώνεται η κατάτμηση που προκύπτει χρησιμοποιώντας αποκλειστικά τον $ED1$ και μειώνεται η ακρίβεια της κατάτμησης όταν χρησιμοποιείται ο δείκτης $ED2$.

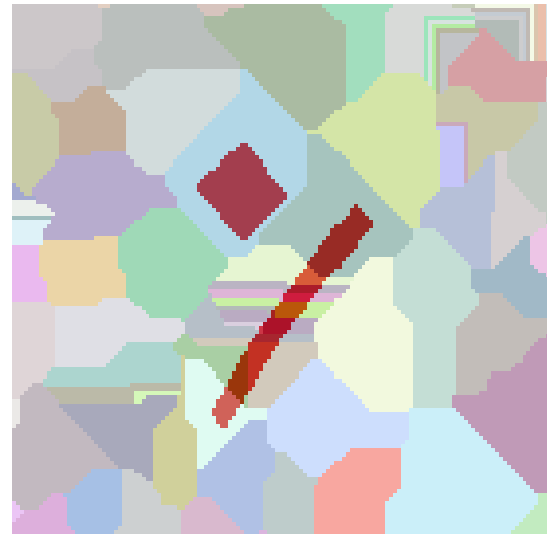
Από την κατάτμηση δημιουργούνται συνολικά 726 αντικείμενα. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση δεν μπορεί να χαρακτηριστεί ως η βέλτιστη δυνατή. Στις εικόνες 3.29 - 3.31 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με τα αντικείμενα αναφοράς. Στην εικόνα 3.29 παρουσιάζεται το σύνολο της εικόνας ενώ στις υπόλοιπες τμηματικά τα αντικείμενα αναφοράς για καλύτερη εποπτική παρουσίαση των αποτελεσμάτων.



Εικόνα 3.29 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη $(ED1+ED2)/2$ (watershed - Min Fitness / Min Raw = 1,14 / 0,60)



Εικόνα 3.30 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.31 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

Προκειμένου να συγκριθεί το αποτέλεσμα της κατάτμησης που προκύπτει με την χρήση του δείκτη ED2, και του δείκτη $(ED2+ED1)/2$ παρουσιάζονται στις εικόνες 3.32 και 3.33 δυο αντίστοιχα τμήματα των κατατμήσεων. Παρατηρείται ότι οι κατατμήσεις διαφέρουν ελάχιστα μεταξύ τους, γεγονός το οποίο οφείλεται στο ότι ο δείκτης $(ED2+ED1)/2$ περιέχει την τιμή του δείκτη ED2. Ωστόσο επειδή το πλήθος και το μέγεθος των αντικειμένων κατάτμησης που περιέχονται ή περιέχουν τα αντικείμενα αναφοράς είναι διαφορετικά στις δυο κατατμήσεις, προκύπτει και η διαφορετική τιμή του δείκτη ED2.



Εικόνα 3.32 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης με δείκτη ED2



Εικόνα 3.33 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης με δείκτη $(ED1+ED2)/2$

Συμπερασματικά παρατηρείται ότι η χρήση του δείκτη ED2 είχε σαν αποτέλεσμα την υλοποίηση καλύτερης κατάτμησης στην εικόνα του Αγ. Στεφάνου, σε σχέση με τους δείκτες ED1 και $(ED1+ED2)/2$. Η τελευταία σε ποιότητα κατάτμηση σε σχέση με τα επιλεγμένα αντικείμενα αναφοράς υλοποιήθηκε με την χρήση του δείκτη ED1. Το γεγονός αυτό επιβεβαιώνει την άποψη ότι ο δείκτης επικεντρώνεται στις γεωμετρικές σχέσεις μεταξύ των αντικειμένων αλλά παραμελεί τις αριθμητικές σχέσεις μεταξύ αυτών, γεγονός που είναι αναγκαίο αλλά όχι αρκετό για να αποδώσει μια βέλτιστη κατάτμηση (Yong Liu et al 2012).

3.5.2.2 Υλοποίηση στην εικόνα από αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας

Τα αποτελέσματα των κατατμήσεων που προέκυψαν χρησιμοποιώντας τον συγκεκριμένο αλγόριθμο κατάτμησης στην υλοποίηση του ΓΑ, εξαρτώνται από την συνάρτηση αξιολόγησης που χρησιμοποιείται κάθε φορά. Καλύτερα αποτελέσματα παρατηρούνται με την χρησιμοποίηση του δείκτη ED2, σε σχέση με τους υπόλοιπους δυο, ως προς το αποτέλεσμα της κατάτμησης. Ωστόσο υπήρξε περίπτωση όπου ένα συγκεκριμένο αντικείμενο εντοπίστηκε καλύτερα με την χρήση του δείκτη ED1 (εικόνα 3.41).

Όσον αφορά τον χρόνο εκτέλεσης του ΓΑ αυτός κυμαίνεται από 24,2 ώρες έως 70,2 ώρες ανάλογα με τις επιλογές του ΓΑ και κυρίως με το πλήθος γενεών και τον εκάστοτε πληθυσμό, χωρίς όμως να αποτελεί βέλτιστη υλοποίηση καμιά από τις δυο εφαρμογές. Για παράδειγμα με χρήση του δείκτη ED2 ως συνάρτηση αξιολόγησης καλύτερα αποτελέσματα προέκυψαν σε 50,7 ώρες στην δεύτερη περίπτωση του πίνακα 3.2.

Όσον αφορά τις επιλογές του ΓΑ στις διαδικασίες υλοποίησης του, επιλέχθηκαν σε κάθε εφαρμογή όσο το δυνατόν διαφορετικές, ώστε να ελεγχθούν τα αποτελέσματα που προκύπτουν. Ωστόσο λόγω του ότι το εύρος τιμών των παραμέτρων δεν ήταν το ίδιο για όλες, δημιούργησε περιορισμό στον τρόπο αρχικοποίησης του αρχικού πληθυσμού καθώς και στον τρόπο διασταύρωσης και μετάλλαξης, καθώς κάποιες μέθοδοι δεν υποστήριζαν την συγκεκριμένη υλοποίηση. Για το λόγο αυτό η αρχικοποίηση και η μετάλλαξη υλοποιήθηκε σε όλες τις εφαρμογές του ΓΑ με την επιλογή «Allele». Ωστόσο οι υπόλοιπες επιλογές του ΓΑ τροποποιούνταν σε κάθε εφαρμογή. Για παράδειγμα ο ρυθμός διασταύρωσης και μετάλλαξης καθώς και το εύρος τιμών των παραμέτρων, όπως φαίνεται και στον πίνακα 3.2.

Η διαφορά μέγιστου και ελάχιστου χρόνου στις εκτελέσεις του ΓΑ οφείλεται στις διαφορετικές διαδικασίες υλοποίησης του ΓΑ σε κάθε εφαρμογή του. Ιδίως το πλήθος του πληθυσμού και των γενεών είναι οι καθοριστικοί παράγοντες του χρόνου εκτέλεσης του ΓΑ. Αν το μέγεθος του πληθυσμού είναι μικρό ελλοχεύει ο κίνδυνος ο ΓΑ να παγιδευτεί σε ένα πλήθος συγκεκριμένων ατόμων, να εντοπίσει το καλύτερο από αυτό και να συνεχίσει την διαδικασία θεωρώντας αυτό ως το καλύτερο δυνατό. Ωστόσο αν υπήρχε μεγαλύτερο πεδίο για έρευνα του ΓΑ, είναι πιθανό τα αποτελέσματα να ήταν διαφορετικά. Γι αυτό το λόγο είναι προτιμότερο να είναι μεγαλύτερο το μέγεθος του πληθυσμού και μικρότερο το πλήθος των γενεών παρά να συμβαίνει το αντίθετο. Για τον λόγο αυτό πρέπει αυτοί να επιλέγονται με ιδιαίτερη προσοχή ώστε να επιτυγχάνονται τα βέλτιστα αποτελέσματα στον συντομότερο χρόνο.

Για στατιστικούς λόγους αναφέρεται ότι ο συνολικός χρόνος εκτέλεσης των επτά εφαρμογών ανέρχεται σε περίπου 309 ώρες με μέσο όρο τις 44 ώρες.

Εικόνα από αερομεταφερόμενο σαρωτή (εταιρία toposys) 473pixel x 474pixel

Ημερ Έναρξης	Χρόνος Εκτ/σης	Pop Size	Initiali- zation	Selection	Crossover	Mutation	Gen	mutation rate	crossover rate	LB (P1_P2_P3), P4	UB (P1_P2_P3) , P4	Συνάρτηση Αξιολόγησης	ΑΠΟΤΕΛΕΣΜΑΤΑ				
													P1	P2	P3	P4	Min Fitness /Min Raw
24/2/13	31,2h	100	Allele (real)	Roulette Wheel	Single Point	Allele	20	2%	90%	1_1_1 3 ή 5 ή 7 ή 9	255_255_15 3 ή 5 ή 7 ή 9	ED2	181	226	3	13	3,4/ 0,53
25/2/13	50,7h	80	Allele (real)	Roulette Wheel	Uniform	Allele	30	3%	80%	0_0_0 3 ή 5 ή 7 ή 9	255_255_10 3 ή 5 ή 7 ή 9	ED2	185	199	2	9	1,5/ 0,43
26/2/13	35,34h	60	Allele (integer)	Tourna- ment	Uniform	Allele	30	1%	80%	1_1_1 3 ή 5 ή 7	255_255_15 3 ή 5 ή 7	ED2	40	194	2	5	0,0/ 0,77
28/2/13	70,2h	60	Allele (integer)	Tourna- ment	Uniform	Allele	60	2,5%	90%	0_0_0 3 ή 5 ή 7 ή 9	255_255_15 3 ή 5 ή 7 ή 9	ED2	167	178	2	3	23,07/ 0,77
28/2/13	47,1h	40	Allele (integer)	Tourna- ment	Uniform	Allele	50	1,5%	85%	1_1_1 3 ή 5 ή 7	255_255_15 3 ή 5 ή 7	ED1	167	250	1	5	0,22/ 0,19
24/2/13	24,2h	60	Allele (integer)	Tourna- ment	Uniform	Allele	15	3%	85%	1_1_1 3 ή 5 ή 7 ή 9	255_255_15 3 ή 5 ή 7 ή 9	(ED1+ED2)/2	225	100	2	9	21/ 0,61
26/2/13	49,6h	80	Allele (integer)	Roulette Wheel	TwoPoint	Allele	20	2%	80%	1_1_1 3 ή 5 ή 7 ή 9	255_255_12 3 ή 5 ή 7 ή 9	(ED1+ED2)/2	39	122	2	3	2,88/ 0,45

Πίνακας 3.2 - Στοιχεία εφαρμογής ΓΑ στην εικόνα από αερομεταφερόμενο σαρωτή σε περιοχή της Γερμανίας, με αλγόριθμο κατάτμησης watershed

Οι βέλτιστες κατατμήσεις που προκύπτουν για την εικόνα, για κάθε μια από τις τρεις συνάρτησης αξιολόγησης, με βάση τα αποτελέσματα του πίνακα 3.2 παρουσιάζονται στις εικόνες παρακάτω. Στην εικόνα 3.34 παρουσιάζονται τα αντικείμενα αναφοράς σε επικάλυψη με την εικόνα από τον αερομεταφερόμενο σαρωτή, για την εποπτική παρουσίαση αυτών.



Εικόνα 3.34 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα σε περιοχή της Γερμανίας

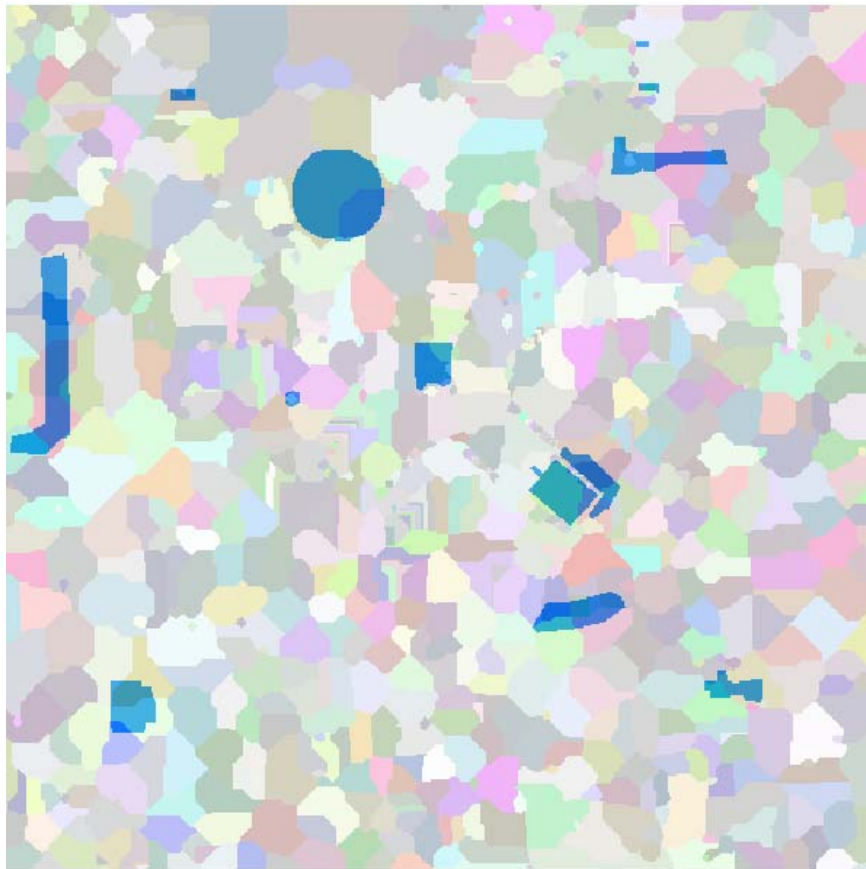
Συνάρτηση αξιολόγησης ED2:

Η χρησιμοποίηση του δείκτη ED2 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED2=0,43$ με τιμές παραμέτρων $P1=185$, $P2=199$, $P3=2$ και $P4=9$. Ωστόσο η τιμή καταλληλότητας δηλαδή η τιμή που προκύπτει σε σύγκριση με τις τιμές του δείκτη για τα υπόλοιπα άτομα είναι 1,5 δηλαδή μεγαλύτερη από την τιμή 0 που προκύπτει για την τρίτη εφαρμογή του ΓΑ με δείκτη ED2, όπως παρουσιάζεται και στον πίνακα 3.2. Για τον λόγο αυτό εξετάζονται τα αποτελέσματα της κατάτμησης και για τις δυο αυτές εφαρμογές του ΓΑ. Για την τιμή $ED2=0,77$ ο δείκτης ED1 λαμβάνει τιμή 0,377 και για την τιμή $ED2=0,43$ ο δείκτης ED1=0,43. Παρατηρείται ότι η τιμή αυτή δεν είναι η βέλτιστη με την χρησιμοποίηση του ED1 ως δείκτη για την συνάρτηση αξιολόγησης, καθώς η βέλτιστη τιμή είναι η $ED1=0,19$. Το γεγονός αυτό ενδυναμώνει το συμπέρασμα ότι οι δυο δείκτες είναι διαφορετικοί μεταξύ τους και επιπλέον ότι ελαχιστοποίηση του ενός δεν συνεπάγεται ταυτόχρονη ελαχιστοποίηση και του άλλου δείκτη.

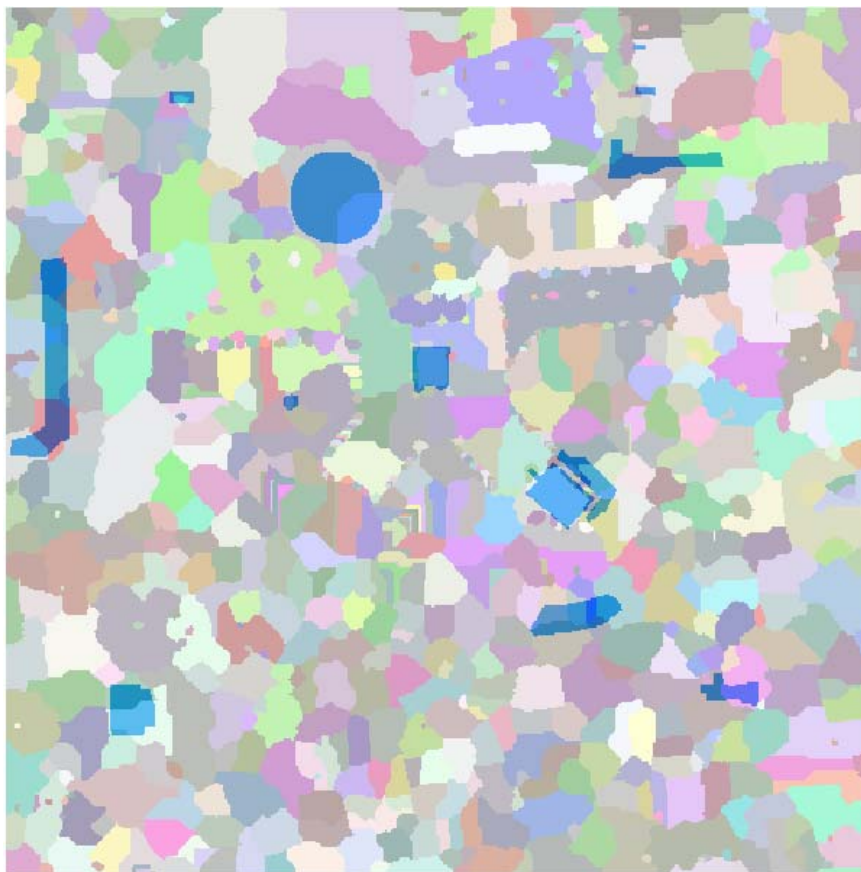
Από την κατάτμηση δημιουργούνται συνολικά 847 αντικείμενα και για τις δυο εφαρμογές του ΓΑ που αναφέρονται. Το γεγονός αυτό αποδεικνύει ο ΓΑ προσπαθώντας να ελαχιστοποιήσει την τιμή του δείκτη ED2, τείνει το πλήθος των αντικειμένων σε ένα συγκεκριμένο αριθμό. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση δεν μπορεί να χαρακτηριστεί ως η βέλτιστη δυνατή ωστόσο η χρήση του ΓΑ απέδωσε ικανοποιητικά αποτελέσματα. Γενικότερα η κατάτμηση με χρήση του αλγορίθμου κατάτμησης με την μέθοδο του υδροκρίτη δεν χαρακτηρίζεται ως βέλτιστη. Στις εικόνες

3.35 έως 3.40 παρουσιάζεται το αποτέλεσμα της κατάτμησης επικαλυπτόμενο από τα αντικείμενα αναφοράς. Στις εικόνες 3.35 και 3.36 παρουσιάζεται το σύνολο των εικόνων κατάτμησης, ενώ στις υπόλοιπες τμηματικά τα αντικείμενα αναφοράς για τις δυο εφαρμογές του ΓΑ, για καλύτερη σύγκριση μεταξύ τους.

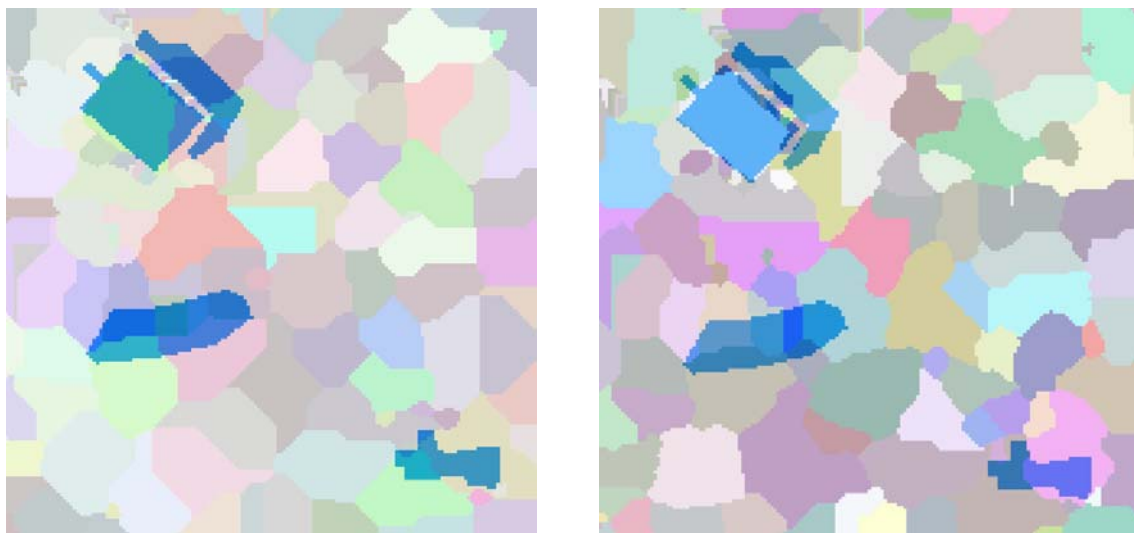
Από τις εικόνες 3.37 έως 3.40 παρατηρείται ότι οι δυο κατατμήσεις είναι πράγματι διαφορετικές μεταξύ τους, οπότε δικαιολογείται και η διαφορά στην τιμή του δείκτη ED2. Καλύτερη κρίνεται η πρώτη κατάτμηση με τιμή δείκτη ED2=0,43 σύμφωνα με την τιμή. Ωστόσο λόγω σύγκρισης των τιμών κάθε ατόμου με τα υπόλοιπα άτομα προκύπτει η τιμή καταλληλότητας η οποία στην δεύτερη κατάτμηση είναι καλύτερη από την πρώτη λόγω του ότι το άτομο που έδωσε τιμή ED2=0,43 δεν περιλαμβάνεται στην δεύτερη κατάτμηση.



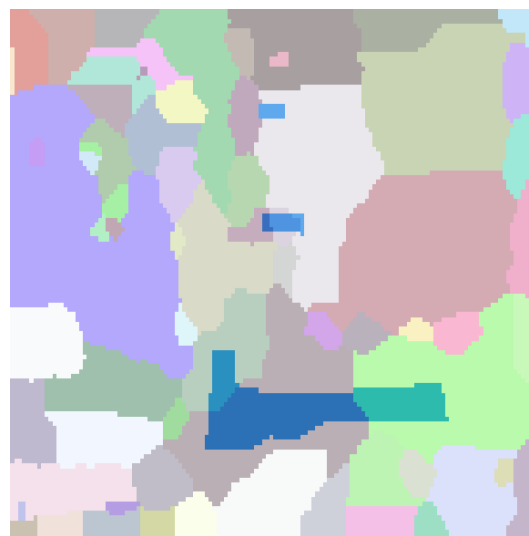
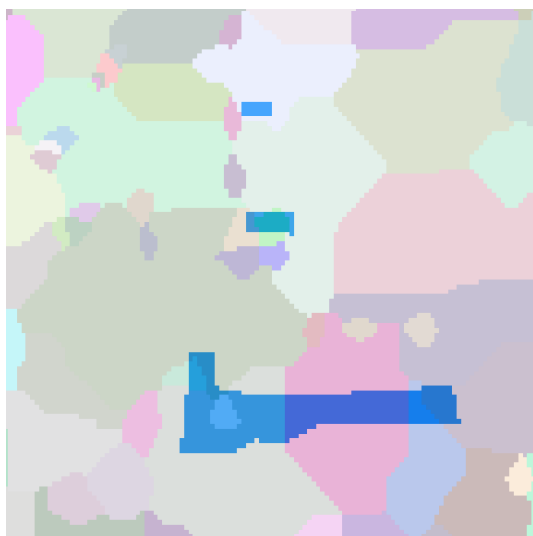
Εικόνα 3.35 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED2 (watershed - Min Fitness / Min Raw = 1,5 / 0,43)



Εικόνα 3.36 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED2 (watershed - Min Fitness / Min Raw = 0,0 / 0,77)



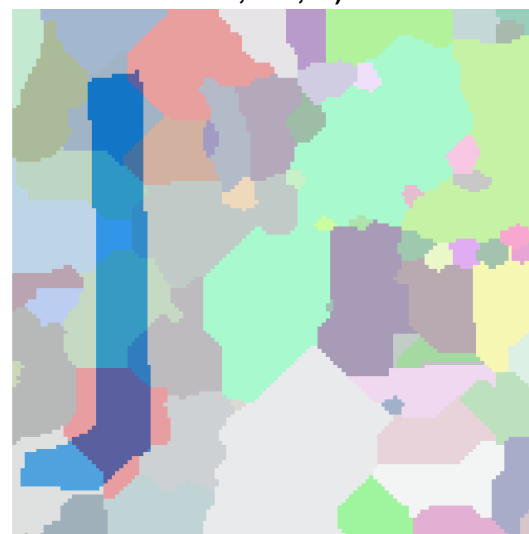
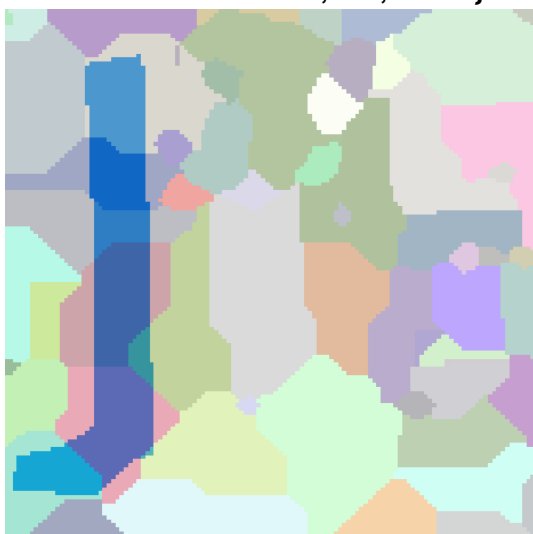
Εικόνα 3.37 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 1,5 / 0,43 - δεξιά Min Fitness / Min Raw = 0,0 / 0,77)



Εικόνα 3.38 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 1,5 / 0,43 - δεξιά Min Fitness / Min Raw = 0,0 / 0,77)



Εικόνα 3.39 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 1,5 / 0,43 - δεξιά Min Fitness / Min Raw = 0,0 / 0,77)

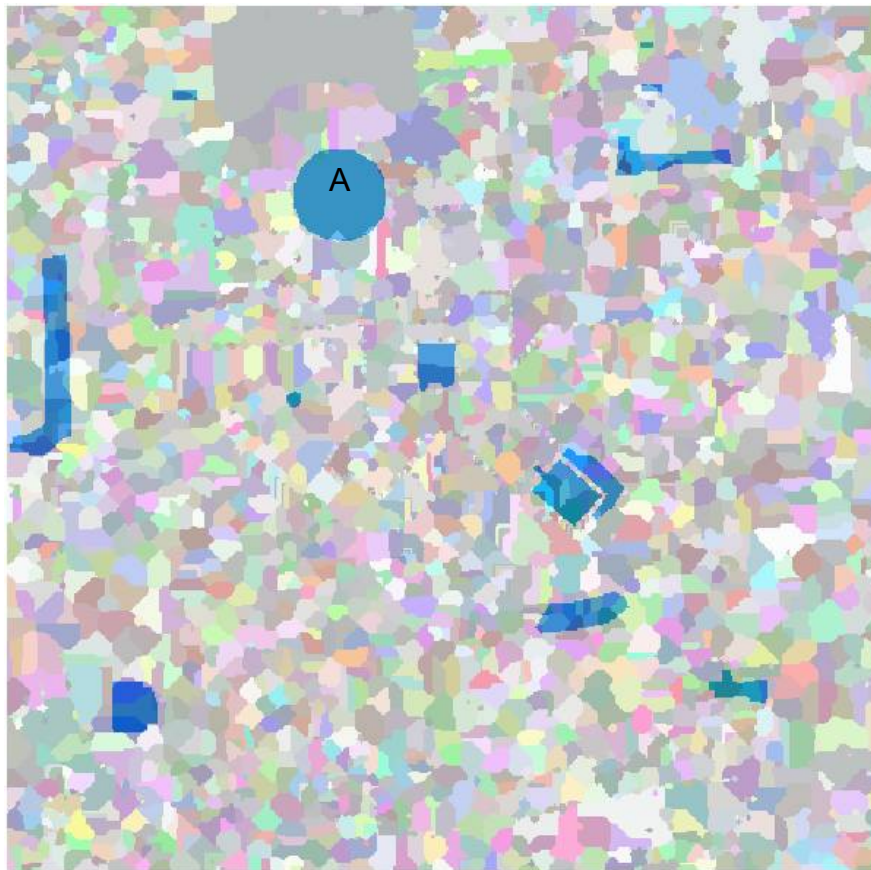


Εικόνα 3.40 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 1,5 / 0,43 - δεξιά Min Fitness / Min Raw = 0,0 / 0,77)

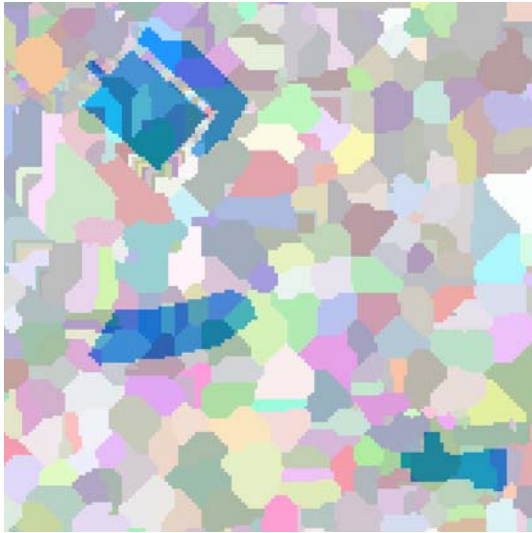
Συνάρτηση αξιολόγησης ED1:

Η χρησιμοποίηση του δείκτη ED1 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED1=0,19$ με τιμές παραμέτρων $P1=167$, $P2=250$, $P3=1$ και $P4=5$. Για τις τιμές αυτές ο δείκτης ED2 λαμβάνει τιμή $ED2=3,39$. Το γεγονός αυτό ενδυναμώνει το συμπέρασμα ότι οι δυο δείκτες είναι διαφορετικοί και ότι ο καθένας από αυτούς συμπεριφέρεται με διαφορετικό τρόπο, καθώς η ελάχιστη τιμή για τον δείκτη ED2 είναι 0,43.

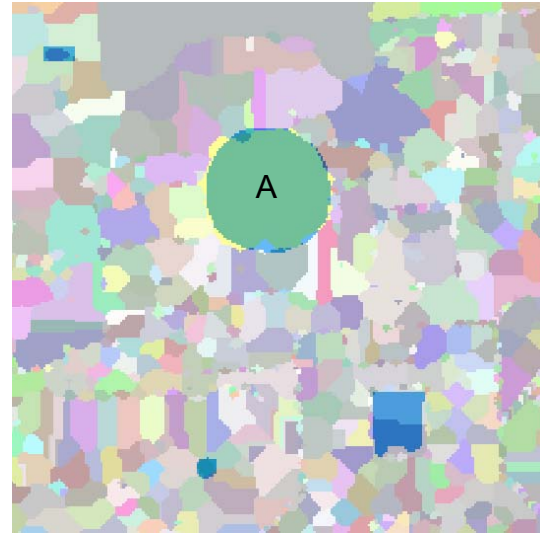
Από την κατάτμηση δημιουργούνται συνολικά 2883 αντικείμενα. Το μέγεθος αυτό είναι ιδιαίτερα μεγάλο και όπως παρουσιάζεται και στις εικόνες 3.41 έως 3.43 η κατάτμηση δεν μπορεί να χαρακτηριστεί ως ορθή, λαμβάνοντας υπόψη τα αντικείμενα αναφοράς, καθώς αυτά περιέχονται σε πάρα πολλά αντικείμενα κατάτμησης. Ωστόσο παρατηρείται ότι υπάρχει ένα αντικείμενο «Α», το οποίο έχει εντοπιστεί ορθά και μάλιστα σε μεγαλύτερο βαθμό από την περίπτωση χρήσης του δείκτη ED2. Ο ΓΑ ελαχιστοποιώντας τον δείκτη ED1, λειτούργησε μεν ορθά όμως η ελαχιστοποίηση του δείκτη δεν αποδίδει τα βέλτιστα αποτελέσματα, καθώς δημιουργείται πλήθος αντικειμένων.



Εικόνα 3.41 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED1 (watershed - Min Fitness / Min Raw = 0,22 / 0,19)



Εικόνα 3.42 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

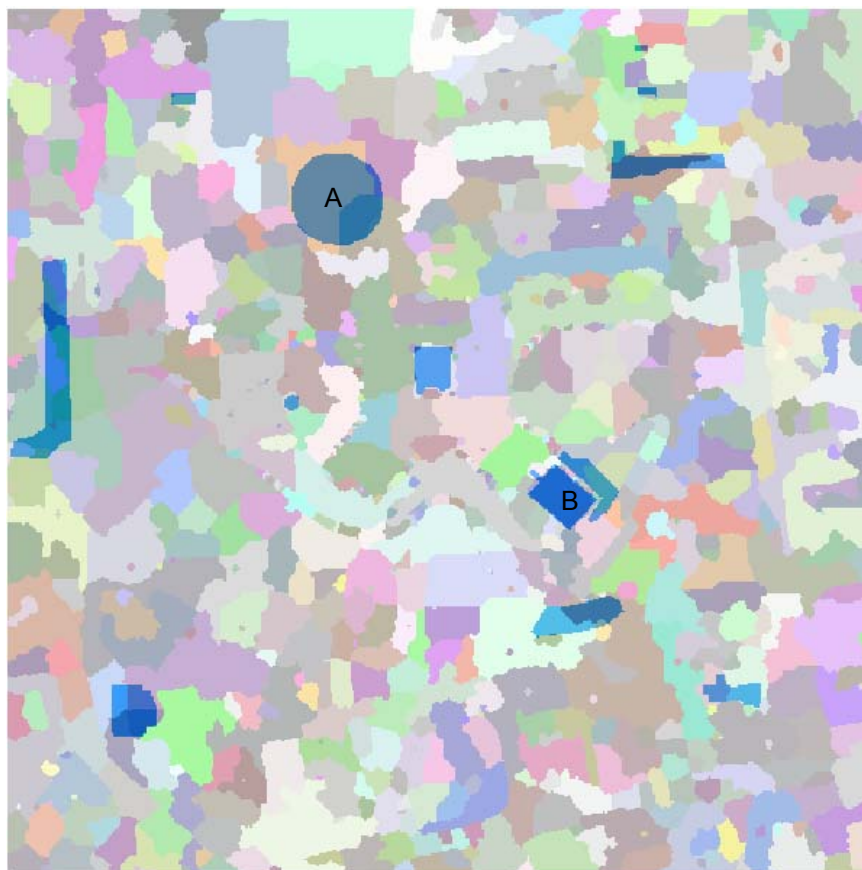


Εικόνα 3.43 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

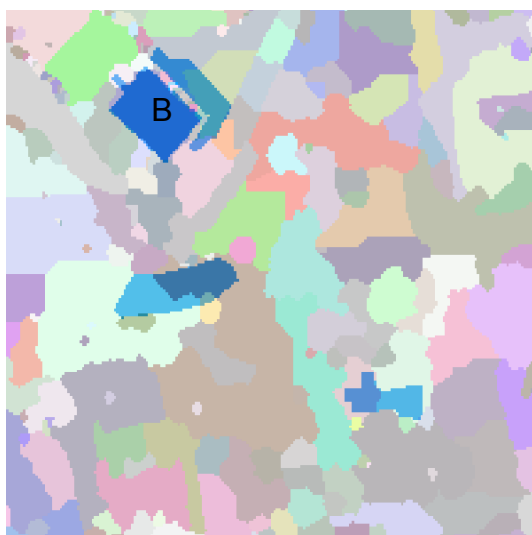
Συνάρτηση αξιολόγησης $(ED1+ED2)/2$:

Η χρησιμοποίηση του δείκτη $(ED1+ED2)/2$ ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή 0,45. Συγκεκριμένα τιμές $ED1=0,35$ και $ED2=0,54$ με τιμές παραμέτρων $P1=39$, $P2=122$, $P3=2$ και $P4=3$. Παρατηρείται ότι με την χρησιμοποίηση του δείκτη αυτού η προκύπτουσα κατάτμηση ακολουθεί κυρίως τον δείκτη $ED2$. Ουσιαστικά βελτιώνεται η κατάτμηση που προκύπτει χρησιμοποιώντας αποκλειστικά τον $ED1$ και μειώνεται η ακρίβεια της κατάτμησης όταν χρησιμοποιείται ο δείκτης $ED2$. Επιπλέον παρατηρείται ότι καμιά από τις τιμές των δυο δεικτών $ED1$ και $ED2$ δεν είναι η ελάχιστη, όμως είναι κοντά στην αντίστοιχη κάθε φορά ελάχιστη. Το γεγονός αυτό αποδεικνύει την ορθή λειτουργία του ΓΑ όπου προσπαθεί να ελαχιστοποιήσει και τις δυο τιμές ταυτόχρονα.

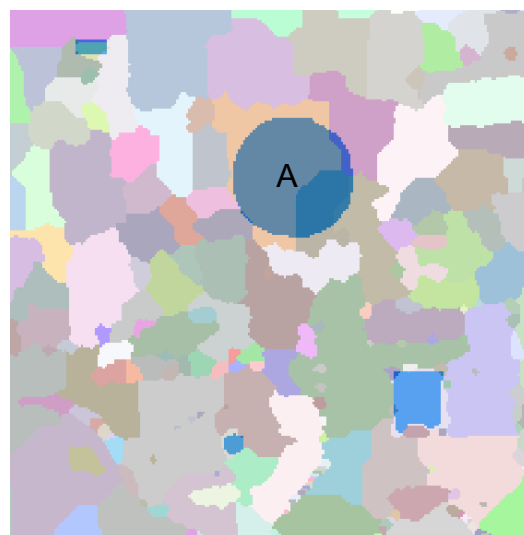
Από την κατάτμηση δημιουργούνται συνολικά 847 αντικείμενα. Το πλήθος αυτό είναι το ίδιο με το πλήθος των αντικειμένων με χρησιμοποίηση του δείκτη $ED2$ ως συνάρτηση αξιολόγησης. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση δεν μπορεί να χαρακτηριστεί ως η βέλτιστη δυνατή. Στις εικόνες 3.44 έως 3.46 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με τα αντικείμενα αναφοράς. Στην εικόνα 3.44 παρουσιάζεται το σύνολο της εικόνας ενώ στις υπόλοιπες τμηματικά τα αντικείμενα αναφοράς για καλύτερη εποπτική παρουσίαση των αποτελεσμάτων. Παρατηρείται ότι το αντικείμενο «B», έχει εντοπιστεί ορθά όπως και στην περίπτωση χρήσης του δείκτη $ED2$, όμως δεν συμβαίνει το ίδιο με το αντικείμενο «A», στην περίπτωση χρήσης του δείκτη $ED1$.



Εικόνα 3.44 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη $(ED1+ED2)/2$ (watershed - Min Fitness / Min Raw = 1,14 / 0,60)



Εικόνα 3.45 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.46 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

3.5.2.3 Υλοποίηση στην ΔΕ της NASA για τον πλανήτη Άρη

Τα αποτελέσματα των κατατμήσεων που προέκυψαν χρησιμοποιώντας τον συγκεκριμένο αλγόριθμο κατάτμησης στην υλοποίηση του ΓΑ, εξαρτώνται από την συνάρτηση αξιολόγησης που χρησιμοποιείται κάθε φορά. Καλύτερα αποτελέσματα παρατηρούνται με την χρησιμοποίηση του δείκτη $(ED1+ED2)/2$, σε σχέση με τους υπόλοιπους δυο, ως προς το αποτέλεσμα της κατάτμησης, γεγονός αντίθετο από τα μέχρι τώρα διαπιστωθέντα. Ακολουθεί ο δείκτης ED2 και στο τέλος έρχεται ο δείκτης ED1.

Όσον αφορά τον χρόνο εκτέλεσης του ΓΑ αυτός κυμαίνεται από 8,2 ώρες έως 42,4 ώρες ανάλογα με τις επιλογές του ΓΑ και κυρίως με το πλήθος γενεών και τον εκάστοτε πληθυσμό. Για παράδειγμα με χρήση του δείκτη ED2 ως συνάρτηση αξιολόγησης καλύτερα αποτελέσματα προέκυψαν σε 13,2 ώρες στην τέταρτη περίπτωση του πίνακα 3.3, ενώ υπήρξε και εφαρμογή με 42,4 ώρες χωρίς ωστόσο να αποδώσει την βέλτιστη κατάτμηση.

Όσον αφορά τις επιλογές του ΓΑ στις διαδικασίες υλοποίησης του, επιλέχτηκαν σε κάθε εφαρμογή όσο το δυνατόν διαφορετικές, ώστε να ελεγχθούν τα αποτελέσματα που προκύπτουν. Ωστόσο λόγω του ότι το εύρος τιμών των παραμέτρων δεν ήταν το ίδιο για όλες, δημιούργησε περιορισμό στον τρόπο αρχικοποίησης του αρχικού πληθυσμού καθώς και στον τρόπο διασταύρωσης και μετάλλαξης, καθώς κάποιες μέθοδοι δεν υποστήριζαν την συγκεκριμένη υλοποίηση. Για το λόγο αυτό η αρχικοποίηση και η μετάλλαξη υλοποιήθηκε σε όλες τις εφαρμογές του ΓΑ με την επιλογή «Allele». Ωστόσο οι υπόλοιπες επιλογές του ΓΑ τροποποιούνταν σε κάθε εφαρμογή. Για παράδειγμα ο ρυθμός διασταύρωσης και μετάλλαξης καθώς και το εύρος τιμών των παραμέτρων, όπως φαίνεται και στον πίνακα 3.3.

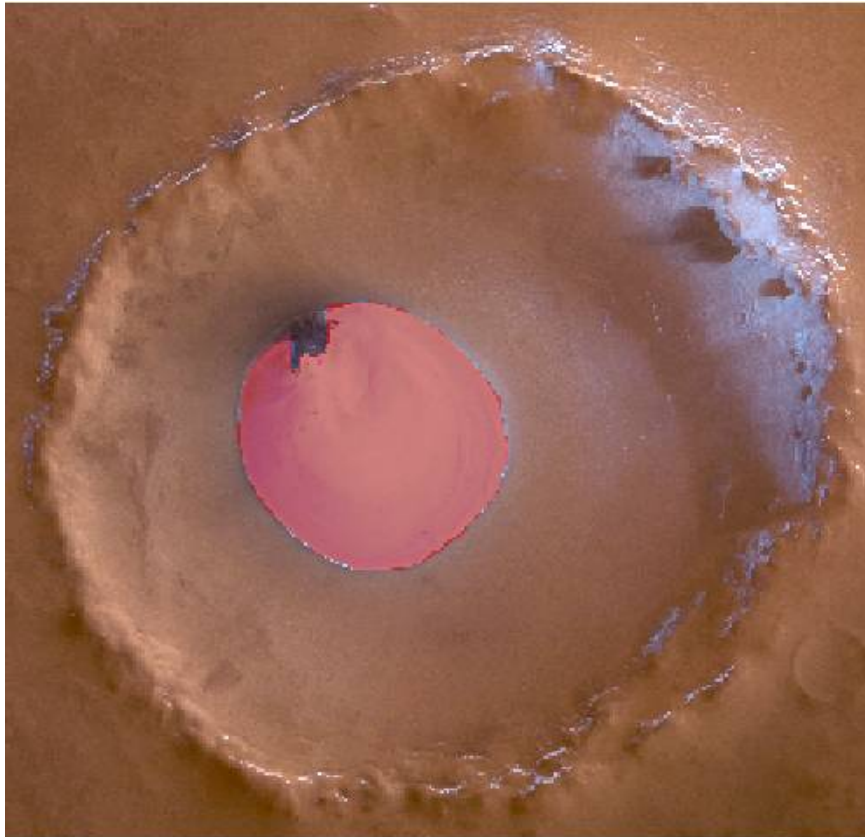
Για στατιστικούς λόγους αναφέρεται ότι ο συνολικός χρόνος εκτέλεσης των εννέα εφαρμογών ανέρχεται σε περίπου 183 ώρες με μέσο όρο τις 20,3 ώρες.

ΔΕ NASA για τον πλανήτη Άρη 299pixel x 307pixel

Ημερ Έναρξης	Χρόνος Εκτ/σης	Pop Size	Initiali- zation	Selection	Crossover	Mutation	Gen	mutation rate	crossover rate	LB (P1_P2_P3), P4	UB (P1_P2_P3) , P4	Συνάρτηση Αξιολόγησης	ΑΠΟΤΕΛΕΣΜΑΤΑ				
													P1	P2	P3	P4	Min Fitness /Min Raw
05/3/13	24,9h	60	Allele (integer)	Tourna- ment	Uniform	Allele	60	2,5%	90%	1_1_1 3 ή 5 ή 7 ή 9	255_255_9 3 ή 5 ή 7 ή 9	ED2	8	245	14	7	24,32/ 0
06/3/13	8,21h	60	Allele (real)	Rank	Uniform	Allele	50	3%	90%	0_0_0 3 ή 5 ή 7 ή 9	255_255_10 3 ή 5 ή 7 ή 9	ED2	187	243	14	7	1,54/ 0,0
06/3/13	42,4h	80	Allele (real)	Roulette Wheel	Uniform	Allele	30	3%	80%	1_1_1 3 ή 5 ή 7 ή 9	255_255_9 3 ή 5 ή 7 ή 9	ED2	185	199	2	9	1,5/ 0,43
07/3/13	13,2h	100	Allele (integer)	Roulette Wheel	TwoPoint	Allele	50	2%	90%	1_1_1 3 ή 5 ή 7 ή 9	255_255_9 3 ή 5 ή 7 ή 9	ED2	199	242	8	5	0,97/ 0,0
19/5/13	11,7h	80	Allele (integer)	Tourna- ment	TwoPoint	Allele	30	3%	80%	1_1_1 3 ή 5 ή 7	255_255_9 3 ή 5 ή 7	ED1	2	138	2	5	0,11/0,02
20/5/13	10,2h	100	Allele (integer)	Uniform	Uniform	Allele	20	2%	90%	1_1_1 3 ή 5 ή 7	255_255_9 3 ή 5 ή 7	ED1	29	140	1	3	0,21/0,02
06/03/13	21,9h	80	Allele (integer)	Roulette Wheel	TwoPoint	Allele	20	2%	85%	1_1_1 3 ή 5 ή 7	255_255_15 3 ή 5 ή 7	(ED1+ED2)/2	22	148	15	3	0,38/ 0,02
07/3/13	28,2h	100	Allele (integer)	Roulette Wheel	Uniform	Allele	15	2%	85%	1_1_1 3 ή 5 ή 7	255_255_15 3 ή 5 ή 7	(ED1+ED2)/2	39	138	6	3	0,32/ 0,017
08/3/13	21,7h	100	Allele (integer)	Tourna- ment	TwoPoint	Allele	30	3%	80%	0_0_0 3 ή 5 ή 7	255_255_10 3 ή 5 ή 7	(ED1+ED2)/2	141	251	8	3	11,97/ 0,017

Πίνακας 3.3 - Στοιχεία εφαρμογής ΓΑ στην ΔΕ της NASA για τον πλανήτη Άρη, με αλγόριθμο κατάτμησης watershed

Οι βέλτιστες κατατμήσεις που προκύπτουν για την εικόνα, για κάθε μια από τις τρεις συνάρτησης αξιολόγησης, με βάση τα αποτελέσματα του πίνακα 3.3 παρουσιάζονται στις εικόνες παρακάτω. Στην εικόνα 3.47 παρουσιάζονται το αντικείμενο αναφοράς σε επικάλυψη με την εικόνα από τον δορυφόρο της NASA, για την εποπτική παρουσίαση αυτού.

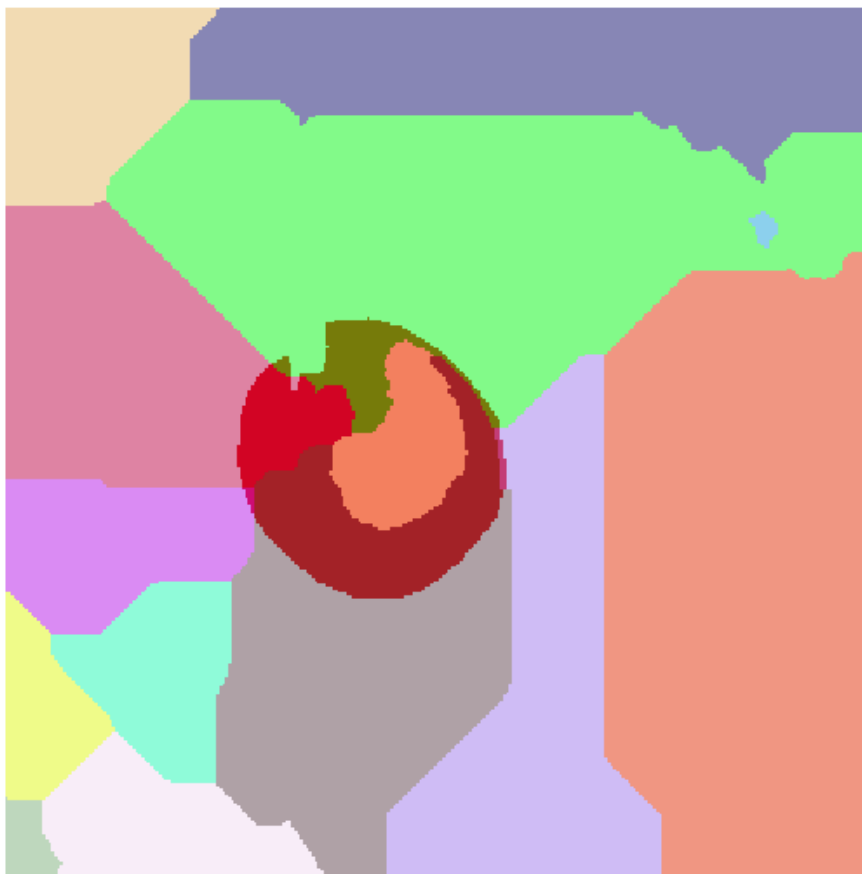


Εικόνα 3.47 - Αντικείμενο αναφοράς σε επικάλυψη με την εικόνα του πλανήτη Άρη

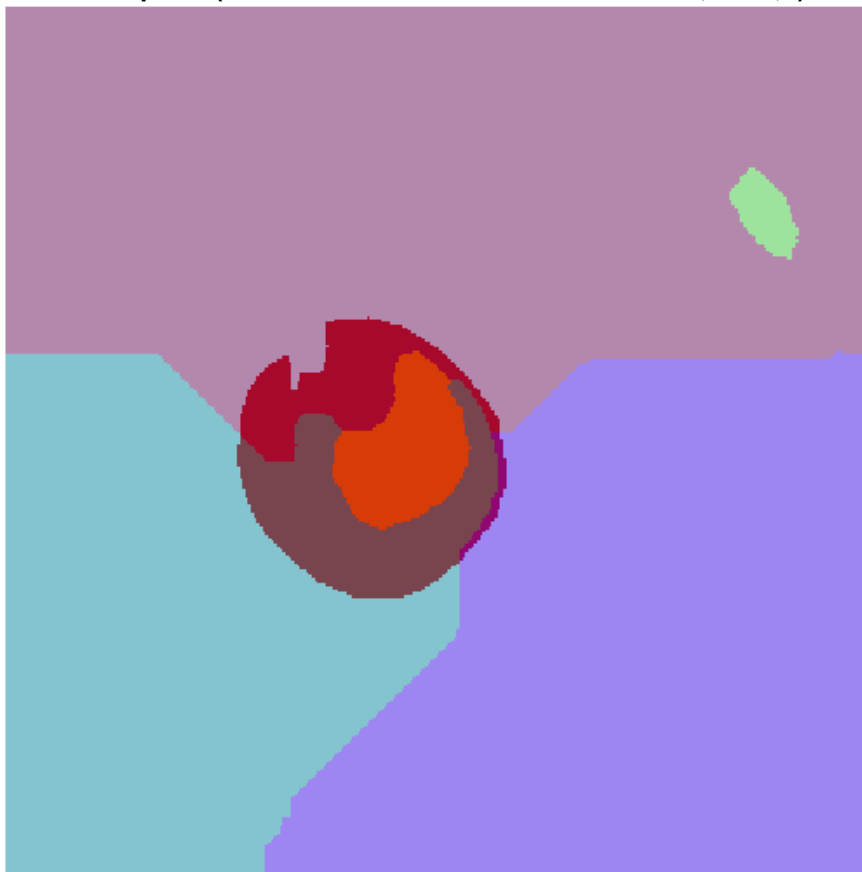
Συνάρτηση αξιολόγησης ED2:

Η χρησιμοποίηση του δείκτη ED2 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED2=0,0$ σε δυο εφαρμογές. Ωστόσο η τιμή καταλληλότητας ήταν διαφορετική σε κάθε μια. Συγκεκριμένα με τιμή καταλληλότητας 0,97 οι τιμές παραμέτρων ήταν $P1=199$, $P2=242$, $P3=8$ και $P4=5$ ενώ για τιμή καταλληλότητας 1,54 οι τιμές των παραμέτρων ήταν $P1=187$, $P2=243$, $P3=14$ και $P4=7$. Ο δείκτης ED1 λαμβάνει τιμή 0.508 στην πρώτη περίπτωση και τιμή 0,521 στην δεύτερη περίπτωση. Παρατηρείται ότι η τιμή αυτή δεν είναι η βέλτιστη με την χρησιμοποίηση του ED1 ως δείκτη για την συνάρτηση αξιολόγησης, καθώς η βέλτιστη τιμή είναι η $ED1=0,02$. Το γεγονός αυτό αποδεικνύει για μια ακόμη φορά την διαφορετικότητα των δυο δεικτών μεταξύ τους.

Από την πρώτη κατάτμηση δημιουργούνται συνολικά 14 αντικείμενα ενώ από την δεύτερη 5. Επιπλέον αν και ο δείκτης ED2 μηδενίστηκε και στις δυο κατατμήσεις, το αντικείμενο αναφοράς δεν εντοπίστηκε σαν ένα αλλά σαν τέσσερα διαφορετικά αντικείμενα και στις δυο κατατμήσεις. Το γεγονός αυτό έχει σαν συμπέρασμα ότι η χρήση του δείκτη ED2 σε εικόνες όπου υπάρχει ένα αντικείμενο δεν λειτουργεί ορθά στην αποτελεσματικότητα της κατάτμησης, με χρήση του συγκεκριμένου αλγορίθμου κατάτμησης. Η συγκεκριμένη κατάτμηση κατ' επέκταση δεν μπορεί να χαρακτηριστεί ως η βέλτιστη δυνατή αν και η χρήση του ΓΑ απέδωσε ικανοποιητικά αποτελέσματα μηδενίζοντας την τιμή της συνάρτησης αξιολόγησης. Στην εικόνα 3.48 παρουσιάζεται το αποτέλεσμα της πρώτης κατάτμησης σε επικάλυψη με το αντικείμενο αναφοράς και στην εικόνα 3.49 το αποτέλεσμα της δεύτερης.



Εικόνα 3.48 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (watershed - Min Fitness / Min Raw = 0,97 / 0,0)

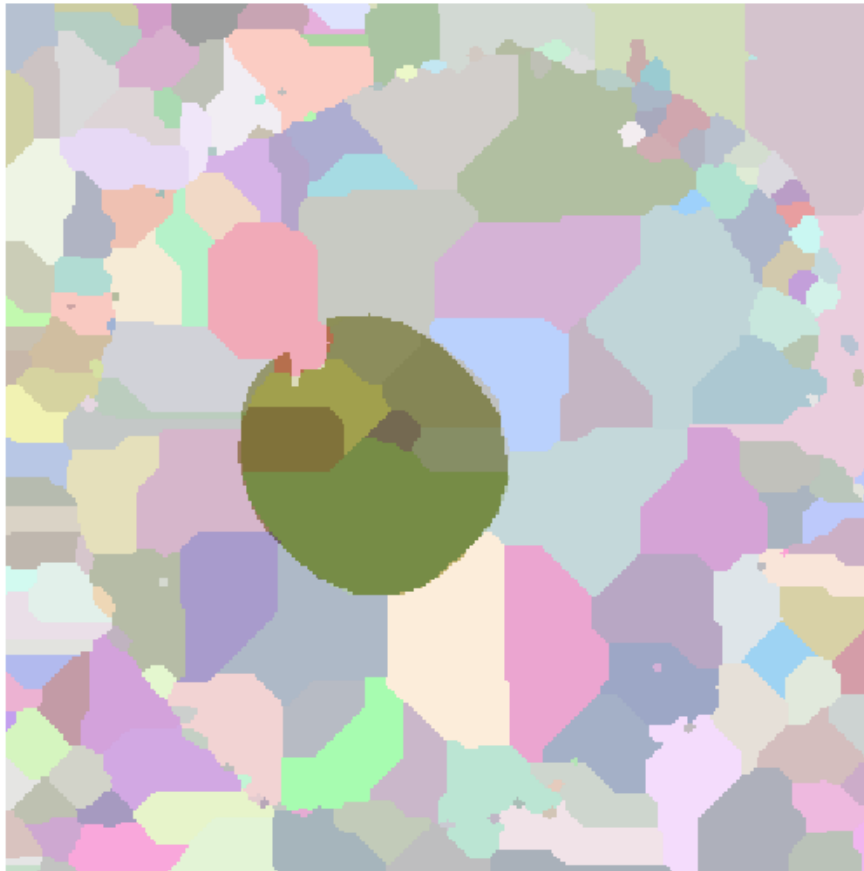


Εικόνα 3.49 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (watershed - Min Fitness / Min Raw = 1,54 / 0,0)

Συνάρτηση αξιολόγησης ED1:

Η χρησιμοποίηση του δείκτη ED1 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED1=0,02$ με τιμές παραμέτρων $P1=22$, $P2=148$, $P3=15$ και $P4=3$. Για τις τιμές αυτές ο δείκτης ED2 λαμβάνει τιμή 6. Η ελάχιστη τιμή του δείκτη ED2, όπως ήδη αναφέρθηκε είναι 0, χωρίς ωστόσο να προκύπτει βέλτιστη κατάτμηση σε αυτή.

Από την κατάτμηση δημιουργούνται συνολικά 207 αντικείμενα. Το μέγεθος αυτό είναι ιδιαίτερα μεγάλο και όπως παρουσιάζεται και στην εικόνα 3.50 η κατάτμηση δεν μπορεί να χαρακτηριστεί ως ορθή, λαμβάνοντας υπόψη το αντικείμενο αναφοράς, καθώς αυτό περιέχεται σε 9 αντικείμενα κατάτμησης, αριθμός μεγαλύτερος από τα 4 αντικείμενα κατάτμησης που συνέθεταν το αντικείμενο αναφοράς με την χρήση του δείκτη ED2. Ο ΓΑ ελαχιστοποιώντας τον δείκτη ED1, λειτούργησε μεν ορθά, όμως η ελαχιστοποίηση του δείκτη δεν αποδίδει τα βέλτιστα αποτελέσματα, καθώς δημιουργείται πλήθος αντικειμένων.

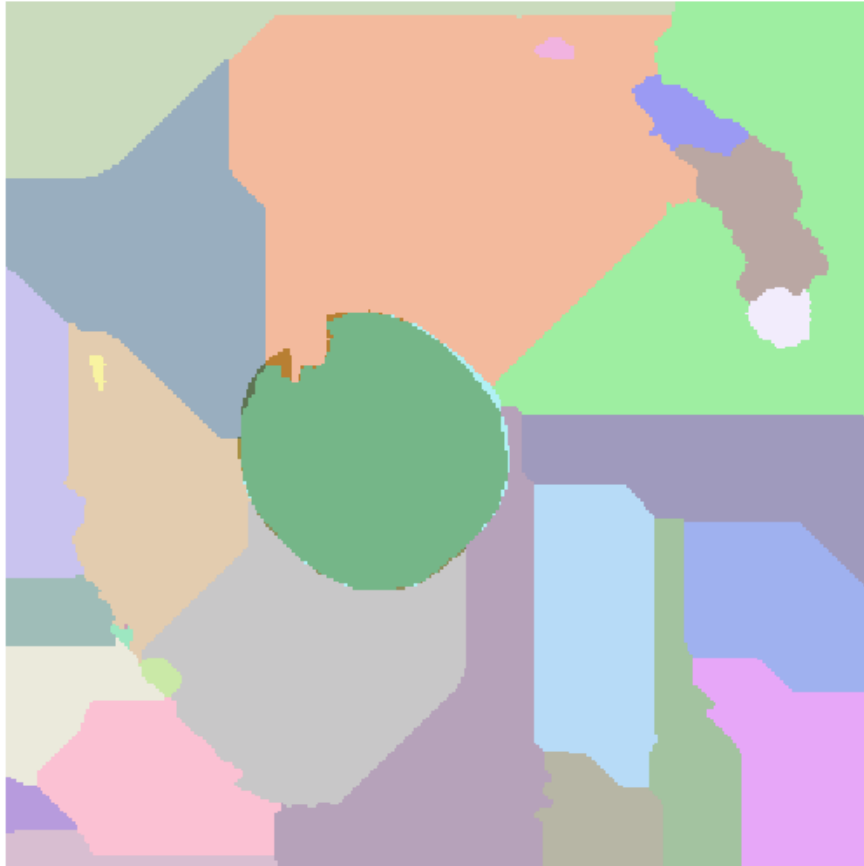


Εικόνα 3.50 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED1 (watershed - Min Fitness / Min Raw = 0,11 / 0,02)

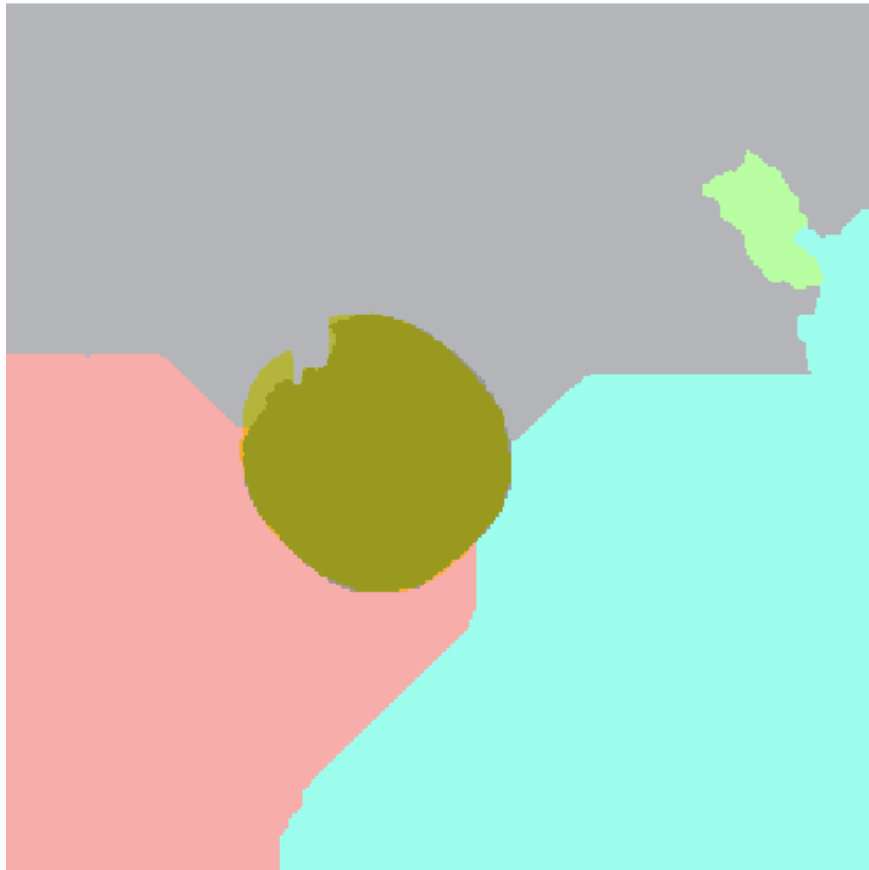
Συνάρτηση αξιολόγησης $(ED1+ED2)/2$:

Η χρησιμοποίηση του δείκτη $(ED1+ED)/2$ ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή 0,017. Συγκεκριμένα τιμές $ED1=0,019$ και $ED2=0,015$ με τιμές παραμέτρων $P1=39$, $P2=138$, $P3=6$ και $P4=3$. Παρατηρείται ότι ο συνδυασμός των δυο δεικτών απέδωσε καλύτερα ως προς τον εντοπισμό του αντικειμένου αναφοράς, τόσο σε σχέση με την αποκλειστική χρήση του δείκτη ED1 όσο και του ED2. Το γεγονός αυτό έρχεται σε αντίθεση με το συμπέρασμα που προέκυψε για τις δυο προηγούμενες εικόνες όπου παρατηρήθηκε ότι η χρήση του συνδυασμένου δείκτη απέφερε ενδιάμεσα και όχι καλύτερα αποτελέσματα.

Από την κατάτμηση δημιουργούνται συνολικά 28 αντικείμενα. Λαμβάνοντας υπόψη το αντικείμενο αναφοράς το οποίο αποτελείται από ένα μόνο αντικείμενο σε ποσοστό άνω του 95% και ότι το πλήθος των αντικειμένων είναι πιο κοντά στην εφαρμογή του δείκτη ED2, και κατ' επέκταση στην πραγματικότητα, η κατάτμηση μπορεί να χαρακτηριστεί ως βέλτιστη. Στις εικόνες 3.51 και 3.52 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με το αντικείμενα αναφοράς. Στην εικόνα 3.51 παρουσιάζεται η κατάτμηση που προκύπτει από την εφαρμογή που αναφέρθηκε, ενώ στην εικόνα 3.52 από την εφαρμογή του ΓΑ όπου έδωσε τιμή δείκτη 0,02. Παρατηρείται ότι στην εικόνα 3.52 το αντικείμενο αναφοράς αποτελείται από ένα αντικείμενο κατάτμησης, ενώ το υπόλοιπο τμήμα της εικόνας αποτελείται από 4 αντικείμενα κατάτμησης, το οποίο παραπέμπει στο δείκτη ED2.



Εικόνα 3.51 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη $(ED1+ED2)/2$ (watershed - Min Fitness / Min Raw = 0,32/0,017)



Εικόνα 3.52 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη $(ED1+ED2)/2$ (watershed - Min Fitness / Min Raw = 0,38 / 0,02)

3.5.3 Υλοποίηση Γενετικού Αλγορίθμου, με Χρησιμοποίηση του Αλγορίθμου Κατάτμησης Region Growing

Στην ενότητα αυτή πραγματοποιείται εφαρμογή του ΓΑ με χρησιμοποίηση του αλγορίθμου κατάτμησης Region Growing. Αρχικά υλοποιήθηκαν πλήθος από δοκιμές στην εικόνα 3.2 (Calw) που ωστόσο λόγω του πειραματικού τους χαρακτήρα αλλά και των μη αναμενόμενων, πολλές φορές, αποτελεσμάτων λόγω διερεύνησης της διαδικασίας δεν αναφέρονται στα εδάφια που ακολουθούν. Στα εδάφια 3.5.3.1, 3.5.3.2 και 3.5.3.3 παρουσιάζονται τα αποτελέσματα για κάθε μια από τις τρεις εικόνες που χρησιμοποιούνται στην διαδικασία υλοποίησης του ΓΑ.

3.5.3.1 Υλοποίηση στην ΔΕ του Αγίου Στεφάνου Αττικής

Τα αποτελέσματα των κατατμήσεων που προέκυψαν χρησιμοποιώντας τον συγκεκριμένο αλγόριθμο κατάτμησης στην υλοποίηση του ΓΑ, εξαρτώνται από την συνάρτηση αξιολόγησης που χρησιμοποιείται κάθε φορά. Καλύτερα αποτελέσματα παρατηρούνται με την χρησιμοποίηση του δείκτη ED2, σε σχέση με τους υπόλοιπους δυο, ως προς το αποτέλεσμα της κατάτμησης ενώ σχεδόν ίδια παρατηρούνται και με την χρήση του δείκτη $(ED1+ED2)/2$.

Όσον αφορά τον χρόνο εκτέλεσης του ΓΑ αυτός κυμαίνεται από 62 ώρες έως 117,3 ώρες ανάλογα με τις επιλογές του ΓΑ και κυρίως με το πλήθος γενεών και τον εκάστοτε πληθυσμό. Ωστόσο όπως παρατηρείται και στον πίνακα 3.4 μεγαλύτερος χρόνος εφαρμογής του αλγορίθμου δεν συνεπάγεται και καλύτερα αποτελέσματα. Για παράδειγμα για την συνάρτηση αξιολόγησης ED2 καλύτερα αποτελέσματα προέκυψαν σε 67,3 ώρες στην πρώτη περίπτωση του πίνακα 3.4, όπου είναι και ο μικρότερος χρόνος εφαρμογής του αλγορίθμου για τον δείκτη ED2.

Όσον αφορά τις επιλογές του ΓΑ στις διαδικασίες υλοποίησης του, επιλέχθηκαν σε κάθε εφαρμογή όσο το δυνατόν διαφορετικές, ώστε να ελεγχθούν τα αποτελέσματα που προκύπτουν. Με την χρήση του συγκεκριμένου αλγόριθμου, καθώς το εύρος τιμών των παραμέτρων ήταν το ίδιο, δόθηκε η δυνατότητα για χρήση περισσότερων επιλογών στην διαδικασία μετάλλαξης του ΓΑ. Ωστόσο στην διαδικασία της αρχικοποίησης εφαρμόστηκε η μέθοδος «Integer», ώστε να δημιουργούνται άτομα με ακέραιες τιμές.

Γενικότερα παρατηρείται ότι για την συγκεκριμένη εικόνα τα αποτελέσματα της κατάτμησης δεν είναι ικανοποιητικά για κανένα δείκτη. Εντοπίζεται με ορθότητα και πληρότητα τρία αντικείμενα χρησιμοποιώντας τον δείκτη ED1 και ένα αντικείμενο για τους υπόλοιπους δυο δείκτες. Επιπλέον παρατηρείται ότι ο χρόνος εκτέλεσης του ΓΑ, όπως και στον προηγούμενο αλγόριθμο, εξαρτάται από τις επιλογές στις διαδικασίες του ΓΑ. Ιδίως το πλήθος του πληθυσμού και των γενεών είναι οι καθοριστικοί παράγοντες του χρόνου εκτέλεσης του ΓΑ. Για τον λόγο αυτό πρέπει αυτοί να επιλέγονται με ιδιαίτερη προσοχή ώστε να επιτυγχάνονται τα βέλτιστα αποτελέσματα στον συντομότερο χρόνο.

Για στατιστικούς λόγους αναφέρεται ότι ο συνολικός χρόνος εκτέλεσης των τεσσάρων εφαρμογών ανέρχεται σε περίπου 331 ώρες με μέσο όρο τις 83 ώρες.

ΔΕ Άγιος Στέφανος 501pixel x 501pixel

Ημερ Έναρξης	Χρόνος Εκτ/σης	Pop Size	Initiali- zation	Selection	Crossover	Mutation	Gen	mutation rate	crossover rate	LB (P1_P2_P3)	UB (P1_P2_P3)	Συνάρτηση Αξιολόγησης	ΑΠΟΤΕΛΕΣΜΑΤΑ				
													P1	P2	P3	conn	Min Fitness /Min Raw
17/5/13	67,3h	40	Integer	Tourna- ment	TwoPoint	Integer Gaussian	10	2%	80%	30	210	ED2	73	33	52	8	126/ 101
25/5/13	117,3h	30	Integer	Roulette Wheel	TwoPoint	Integer Range	15	2%	90%	30	230	ED2	194	46	67	8	129/ 101
25/3/13	84,5h	40	Integer	Tourna- ment	TwoPoint	Integer Gaussian	10	2%	85%	20	220	ED1	47	143	205	8	0,43/ 0,70
20/3/13	62h	40	Integer	Tourna- ment	Uniform	SIM	10	2%	80%	30	230	(ED1+ED2)/2	209	57	84	8	67/ 51

Πίνακας 3.4 - Στοιχεία εφαρμογής ΓΑ στην ΔΕ ΙΚΟΝΟΣ περιοχής του Αγίου Στεφάνου Αττικής, με αλγόριθμο κατάμησης region growing

Οι βέλτιστες κατατμήσεις που προκύπτουν για την εικόνα, για κάθε μια από τις τρεις συνάρτησης αξιολόγησης, με βάση τα αποτελέσματα του πίνακα 3.4 παρουσιάζονται στις εικόνες παρακάτω. Στην εικόνα 3.53 παρουσιάζονται τα αντικείμενα αναφοράς τα οποία επικαλύπτουν την ΔΕ του Αγίου Στεφάνου, για την εποπτική παρουσίαση αυτών.

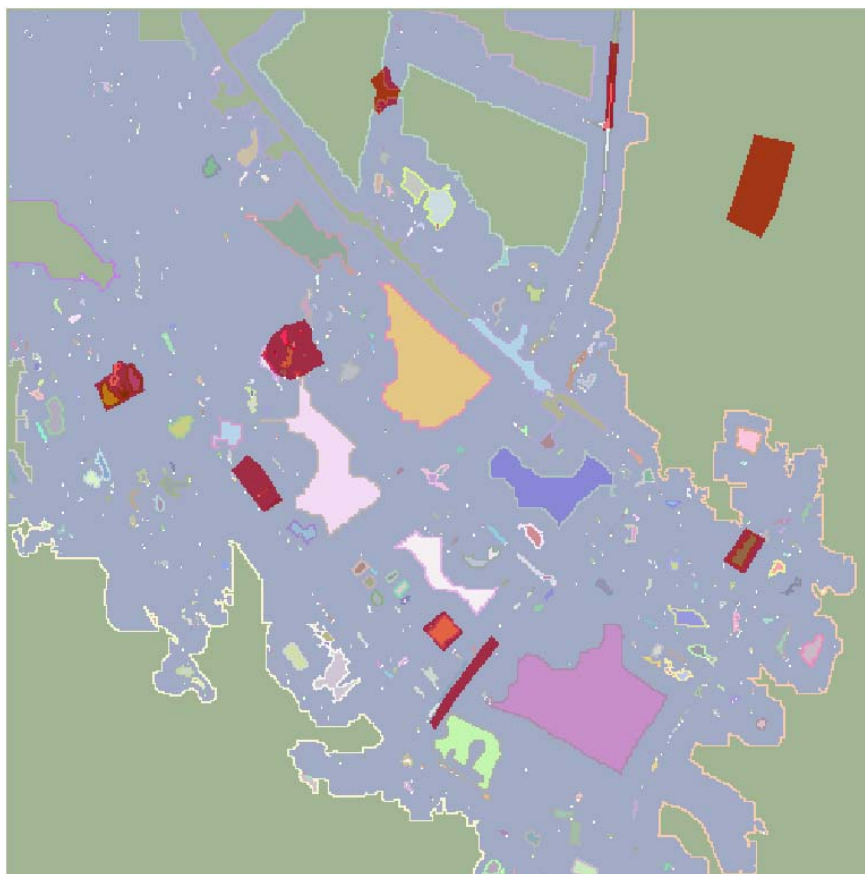


Εικόνα 3.53 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα του Αγ. Στεφάνου

Συνάρτηση αξιολόγησης ED2:

Η χρησιμοποίηση του δείκτη ED2 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED2=101$ με τιμές παραμέτρων $P1=73$, $P2=33$ και $P3=52$. Για τις τιμές αυτές ο δείκτης ED1 λαμβάνει τιμή $ED1=0,717$. Παρατηρείται ότι η τιμή αυτή δεν είναι η βέλτιστη με την χρησιμοποίηση του ED1 ως δείκτη για την συνάρτηση αξιολόγησης, αλλά είναι πολύ κοντά στην βέλτιστη (0,70). Το γεγονός αυτό αποδεικνύει ότι οι δυο δείκτες είναι διαφορετικοί μεταξύ τους και με την χρησιμοποίηση του συγκεκριμένου αλγορίθμου κατάτμησης και συγκεκριμένα ελαχιστοποίηση του ενός δεν συνεπάγεται ταυτόχρονη ελαχιστοποίηση και του άλλου δείκτη.

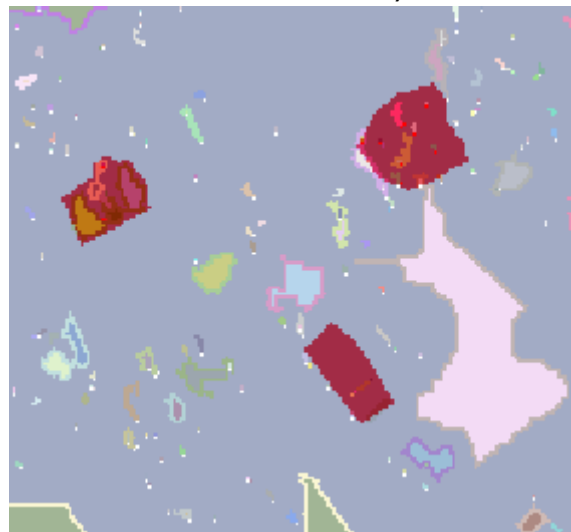
Από την κατάτμηση δημιουργούνται συνολικά 775 αντικείμενα. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση δεν μπορεί να χαρακτηριστεί ως ικανοποιητική. Επιπλέον η χρήση του ΓΑ δεν μπορεί να θεωρηθεί ότι απέδωσε βέλτητα αποτελέσματα καθώς η τελική τιμή του δείκτη είναι ιδιαίτερα υψηλή. Στις εικόνες 3.54 έως 3.58 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με τα αντικείμενα αναφοράς. Στην εικόνα 3.54 παρουσιάζεται το σύνολο της εικόνας ενώ στις υπόλοιπες τμηματικά μέρος των αντικειμένων αναφοράς, με τα καλύτερα αποτελέσματα, για καλύτερη εποπτική παρουσίαση των αποτελεσμάτων σε σχέση με τα αντικείμενα αναφοράς.



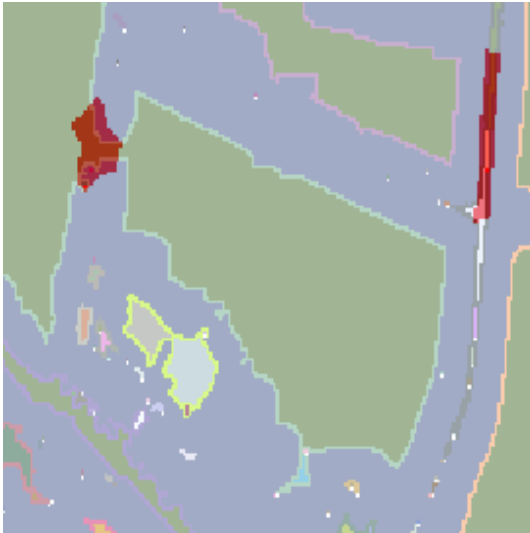
Εικόνα 3.54 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED2 (Region Growing - Min Fitness / Min Raw = 126 / 101)



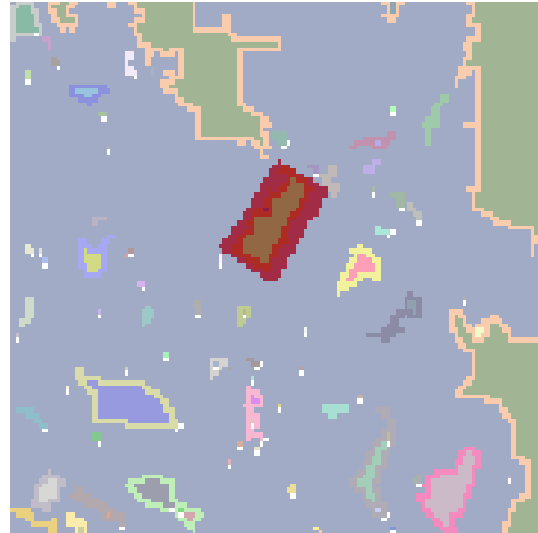
Εικόνα 3.55 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.56 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.57- Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.58 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

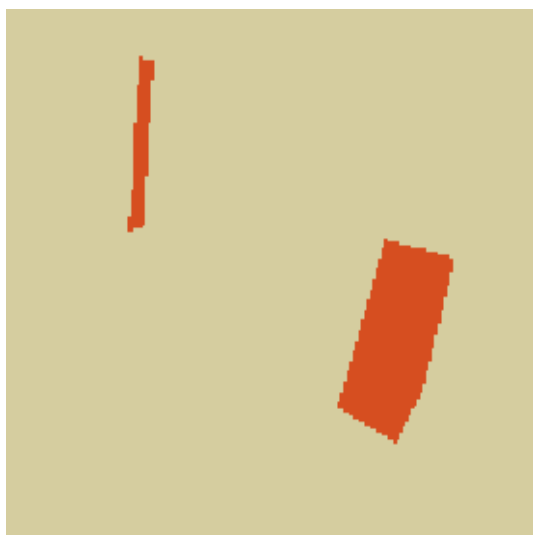
Συνάρτηση αξιολόγησης ED1:

Η χρησιμοποίηση του δείκτη ED1 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED1=0,703$ με τιμές παραμέτρων $P1=47$, $P2=143$ και $P3=205$. Για τις τιμές αυτές ο δείκτης ED2 λαμβάνει τιμή $ED=115,254$. Παρατηρείται ότι ο δείκτης αυτός είναι πλησίον του βέλτιστου όμως υπάρχει μια μικρή διαφορά. Το γεγονός αυτό είναι αντίστοιχο με την εξέταση της περίπτωσης με συνάρτηση αξιολόγησης τον δείκτη ED2.

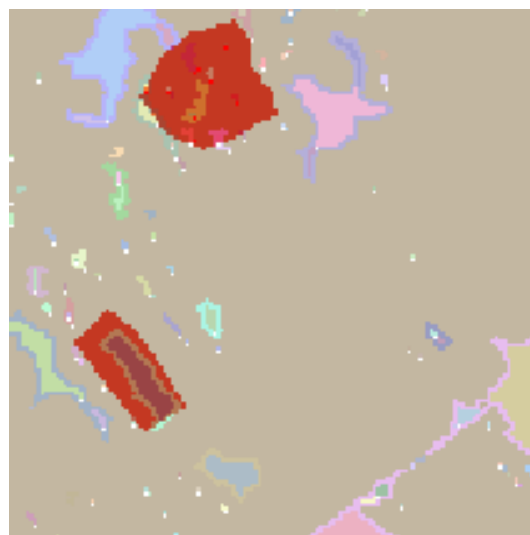
Από την κατάτμηση δημιουργούνται συνολικά 1021 αντικείμενα. Το μέγεθος αυτό είναι μεγαλύτερο από το αντίστοιχο με δείκτη αξιολόγησης τον ED2. Όπως παρουσιάζεται και στις εικόνες 3.59 έως 3.63 η κατάτμηση δεν μπορεί να χαρακτηριστεί ως ικανοποιητική, αντίστοιχα με προηγούμενως. Τα αντικείμενα αναφοράς δεν έχουν εντοπιστεί ορθά, κάποια από αυτά αποτελούν τμήματα ενός πολύ μεγαλύτερου αντικειμένου (εικόνα 3.60), ενώ κάποια από αυτά περιέχονται ή περιέχουν αριθμό αντικειμένων κατάτμησης (εικόνα 3.61 και 3.62). Ωστόσο υπάρχουν δυο αντικείμενα τα οποία εντοπίστηκαν ορθά (εικόνα 3.63).



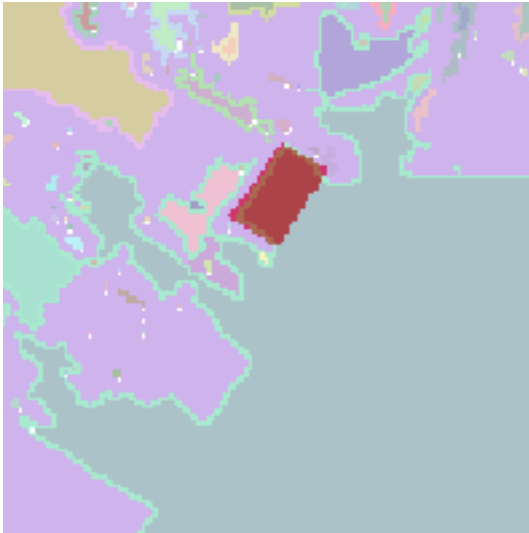
Εικόνα 3.59 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED1 (Region Growing - Min Fitness / Min Raw = 0,43 / 0,70)



Εικόνα 3.60 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.61 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.62 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.63 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

Συνάρτηση αξιολόγησης $(ED1+ED2)/2$:

Η χρησιμοποίηση του δείκτη $(ED1+ED)/2$ ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή 51. Συγκεκριμένα τιμές $ED1=0,717$ και $ED2=101$ με τιμές παραμέτρων $P1=209$, $P2=57$ και $P3=84$. Παρατηρείται ότι με την χρησιμοποίηση του δείκτη αυτού η προκύπτουσα κατάτμηση ακολουθεί και τους δυο δείκτες και σχεδόν πετυχαίνει την βελτιστοποίηση και των δυο σε σχέση με τις βέλτιστες τιμές που επιτεύχθηκαν προηγουμένως για αυτούς. Ωστόσο παρατηρήθηκε ήδη ότι η βελτιστοποίηση του ενός δείκτη οδηγούσε σχεδόν στην βελτιστοποίηση και του άλλου οπότε είναι λογικό ο δείκτης αυτός να οδηγήσει στην βελτιστοποίηση και των δυο ταυτόχρονα.

Από την κατάτμηση δημιουργούνται συνολικά 775 αντικείμενα. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση δεν μπορεί να χαρακτηριστεί ως ικανοποιητική. Από την εικόνα 3.64 παρατηρείται ότι η κατάτμηση ακολουθεί κυρίως την αντίστοιχη με χρησιμοποίηση του δείκτη $ED2$ και όχι του $ED1$. Στις εικόνες 3.64 έως 3.66 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με τα αντικείμενα αναφοράς. Στην εικόνα 3.64 παρουσιάζεται το σύνολο της εικόνας ενώ στις υπόλοιπες τμηματικά τα αντικείμενα αναφοράς για καλύτερη εποπτική παρουσίαση των αποτελεσμάτων. Παρατηρείται η αντιστοιχία στο αποτέλεσμα της κατάτμησης με τις αντίστοιχες εικόνες με χρήση του δείκτη $ED2$ (εικόνες 3.54 έως 3.56).



Εικόνα 3.64 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη $(ED1+ED2)/2$ (Region Growing - Min Fitness / Min Raw = 67 / 51)



Εικόνα 3.65 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.66 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

3.5.3.2 Υλοποίηση στην εικόνα από αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας

Τα αποτελέσματα των κατατμήσεων που προέκυψαν χρησιμοποιώντας τον συγκεκριμένο αλγόριθμο κατάτμησης στην υλοποίηση του ΓΑ δεν εξαρτώνται σε μεγάλο βαθμό από την συνάρτηση αξιολόγησης που χρησιμοποιείται κάθε φορά, σε αντίθεση με τα μέχρι τώρα αναφερθέντα. Τα αποτελέσματα που προκύπτουν με την χρήση και των τριών δεικτών είναι σχεδόν παρόμοια.

Όσον αφορά τον χρόνο εκτέλεσης του ΓΑ αυτός κυμαίνεται από 68,9 ώρες έως 144,6 ώρες ανάλογα με τις επιλογές του ΓΑ και κυρίως με το πλήθος γενεών και τον εκάστοτε πληθυσμό. Με χρήση του δείκτη ED1 ως συνάρτηση αξιολόγησης καλύτερα αποτελέσματα προέκυψαν σε 68,9 ώρες στην τέταρτη περίπτωση του πίνακα 3.5, παρά το ότι στην τρίτη περίπτωση ο χρόνος εκτέλεσης ήταν μεγαλύτερος, καθώς το μέγεθος του πληθυσμού ήταν μεγαλύτερο.

Όσον αφορά τις επιλογές του ΓΑ στις διαδικασίες υλοποίησης του, επιλέχτηκαν σε κάθε εφαρμογή όσο το δυνατόν διαφορετικές, ώστε να ελεγχθούν τα αποτελέσματα που προκύπτουν. Με την χρήση του συγκεκριμένου αλγόριθμου, καθώς το εύρος τιμών των παραμέτρων ήταν το ίδιο, δόθηκε η δυνατότητα για χρήση περισσότερων επιλογών στην διαδικασία μετάλλαξης του ΓΑ. Ωστόσο στην διαδικασία της αρχικοποίησης εφαρμόστηκε η μέθοδος «Integer», ώστε να δημιουργούνται άτομα με ακέραιες τιμές, με εξαίρεση μια εφαρμογή όπου χρησιμοποιήθηκε η δυνατότητα «Allele».

Γενικότερα παρατηρείται ότι για την συγκεκριμένη εικόνα τα αποτελέσματα της κατάτμησης δεν είναι βέλτιστα για κανένα δείκτη, ωστόσο μπορούν να κριθούν ικανοποιητικά. Εντοπίζεται με ορθότητα και πληρότητα ελάχιστα αντικείμενα με την χρησιμοποίηση και των τριών δεικτών. Επιπλέον παρατηρείται ότι ο χρόνος εκτέλεσης του ΓΑ, όπως και στον προηγούμενο αλγόριθμο, εξαρτάται από τις επιλογές στις διαδικασίες του ΓΑ. Ιδίως το πλήθος του πληθυσμού και των γενεών είναι οι καθοριστικοί παράγοντες του χρόνου εκτέλεσης του ΓΑ. Για τον λόγο αυτό πρέπει αυτοί να επιλέγονται με ιδιαίτερη προσοχή ώστε να επιτυγχάνονται τα βέλτιστα αποτελέσματα στον συντομότερο χρόνο.

Για στατιστικούς λόγους αναφέρεται ότι ο συνολικός χρόνος εκτέλεσης των πέντε εφαρμογών ανέρχεται σε περίπου 551 ώρες με μέσο όρο τις 110 ώρες.

Εικόνα από αερομεταφερόμενο σαρωτή (εταιρία toposys) 473pixel x 474pixel

Ημερ Έναρξης	Χρόνος Εκτ/σης	Pop Size	Initiali- zation	Selection	Crossover	Mutation	Gen	mutation rate	crossover rate	LB (P1_P2_P3)	UB (P1_P2_P3)	Συνάρτηση Αξιολόγησης	ΑΠΟΤΕΛΕΣΜΑΤΑ				
													P1	P2	P3	conn	Min Fitness /Min Raw
03/5/13	144,6h	80	Allele (real)	Tourna- ment	Two Point	SIM	10	2%	90%	10_30_30	220_200_200	ED2	67	189	50	8	131/ 42,86
22/5/13	118,6h	40	Integer	Tourna- ment	Single Point	Integer Gaussian	10	2%	80%	30	200	ED2	95	42	163	8	42,1/ 49,2
23/5/13	94,2h	40	Integer	Uniform	Two Point	SIM	10	3%	90%	50	200	ED1	123	149	129	8	0,62 / 0,70
26/5/13	68,9h	20	Integer	Tournam- ent	Uniform	Integer Range	10	2%	85%	30	200	ED1	53	31	147	8	0,61 / 0,70
25/5/13	126h	30	Integer	Roulette Wheel	Uniform	Integer Range	15	3%	85%	20	200	(ED1+ED2)/2	147	172	50	8	22 / 21

Πίνακας 3.5 - Στοιχεία εφαρμογής ΓΑ στην εικόνα από αερομεταφερόμενο σαρωτή σε περιοχή της Γερμανίας, με αλγόριθμο κατάτμησης region growing

Οι βέλτιστες κατατμήσεις που προκύπτουν για την εικόνα, για κάθε μια από τις τρεις συνάρτησης αξιολόγησης, με βάση τα αποτελέσματα του πίνακα 3.5 παρουσιάζονται στις εικόνες παρακάτω. Στην εικόνα 3.67 παρουσιάζονται τα αντικείμενα αναφοράς σε επικάλυψη με την εικόνα από τον αερομεταφερόμενο σαρωτή, για την εποπτική παρουσίαση αυτών.



Εικόνα 3.67 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα σε περιοχή της Γερμανίας

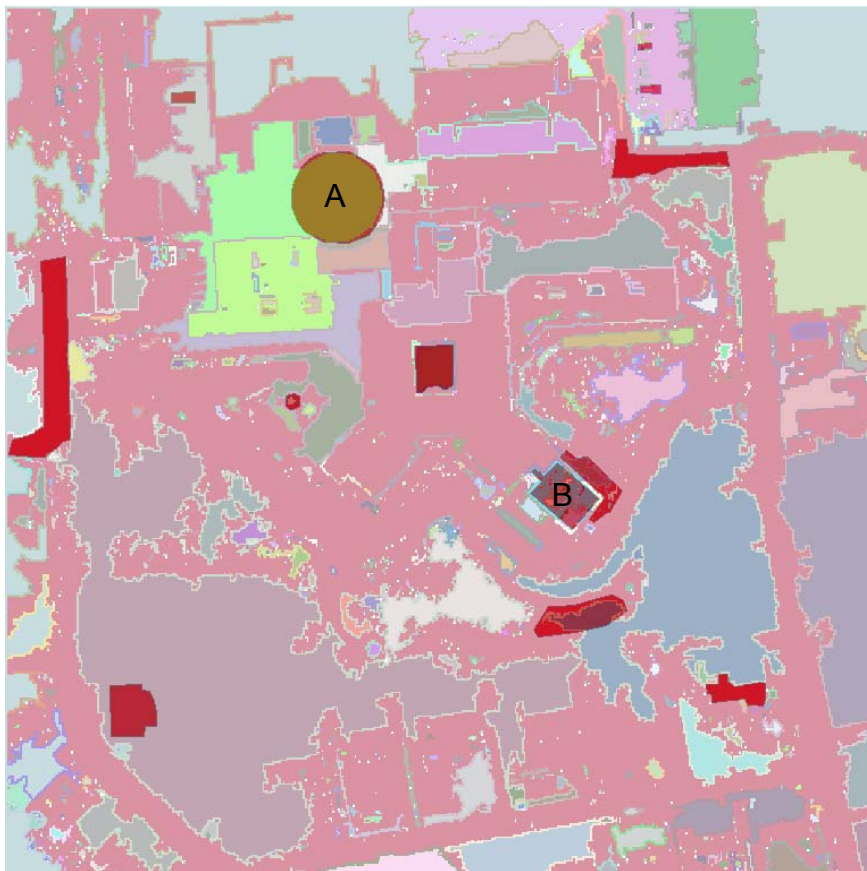
Συνάρτηση αξιολόγησης ED2:

Η χρησιμοποίηση του δείκτη ED2 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED2=42,86$ με τιμές παραμέτρων $P1=67$, $P2=189$ και $P3=50$. Ωστόσο η τιμή καταλληλότητας δηλαδή η τιμή που προκύπτει σε σύγκριση με τις τιμές του δείκτη για τα υπόλοιπα άτομα είναι 131 δηλαδή μεγαλύτερη από την τιμή 42,1 που προκύπτει για την δεύτερη εφαρμογή του ΓΑ με δείκτη $ED2(\text{raw})=49,2$, όπως παρουσιάζεται και στον πίνακα 3.5. Για τον λόγο αυτό εξετάζονται τα αποτελέσματα της κατάτμησης και για τις δυο αυτές εφαρμογές του ΓΑ. Για την τιμή $ED2=42,86$ ο δείκτης ED1 λαμβάνει τιμή 0,702 και για την τιμή $ED2=49,2$ ο δείκτης ED1=0,701. Παρατηρείται ότι η τιμή αυτή του δείκτη ED1 και για τις δυο εφαρμογές είναι η βέλτιστη με την χρησιμοποίηση του ED1 ως δείκτη για την συνάρτηση αξιολόγησης. Καθώς ο ΓΑ μειώνει την τιμή του δείκτη ED2, παράλληλα μειώνει και την τιμή του δείκτη ED1. Οι δυο δείκτες δρουν κατ' επέκταση σχεδόν με τον ίδιο τρόπο.

Από την κατάτμηση δημιουργούνται συνολικά 1999 αντικείμενα για την πρώτη εφαρμογή και 1964 για την δεύτερη. Η διαφορά είναι πολύ μικρή. Το γεγονός αυτό αποδεικνύει ότι ο ΓΑ προσπαθώντας να ελαχιστοποιήσει την τιμή του δείκτη ED2, τείνει το πλήθος των αντικειμένων σε ένα συγκεκριμένο αριθμό. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση δεν μπορεί να χαρακτηριστεί ως η βέλτιστη δυνατή ωστόσο η χρήση του ΓΑ απέδωσε ικανοποιητικά αποτελέσματα. Στις εικόνες 3.68 έως 3.72 παρουσιάζεται το

αποτέλεσμα της κατάτμησης επικαλυπτόμενο από τα αντικείμενα αναφοράς. Στις εικόνες 3.68 και 3.69 παρουσιάζεται το σύνολο των εικόνων κατάτμησης, ενώ στις υπόλοιπες τμηματικά τα αντικείμενα αναφοράς για τις δυο εφαρμογές του ΓΑ, για καλύτερη σύγκριση μεταξύ τους.

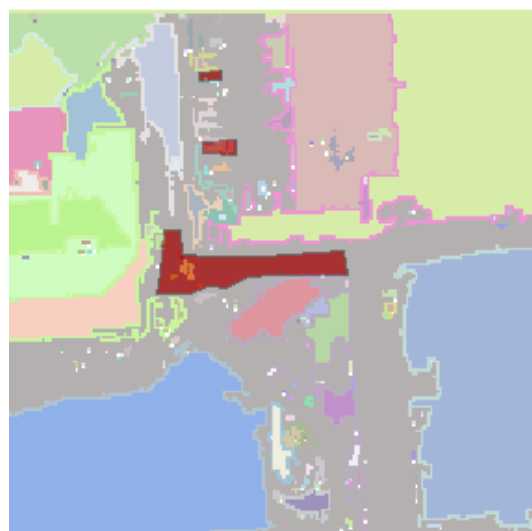
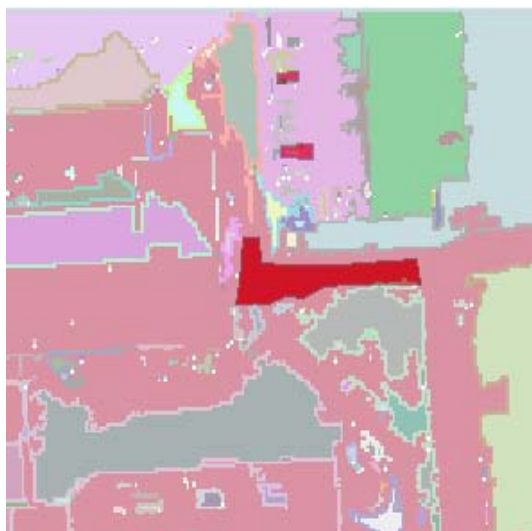
Από τις εικόνες 3.70 έως 3.72 παρατηρείται ότι οι δυο κατατμήσεις είναι πράγματι διαφορετικές μεταξύ τους, οπότε δικαιολογείται και η διαφορά στην τιμή του δείκτη ED2. Παρατηρείται ότι στην πρώτη κατάτμηση το αντικείμενο «Α» εντοπίζεται σωστά ενώ στην δεύτερη όχι. Ωστόσο στην δεύτερη κατάτμηση του αντικείμενο «Β» εντοπίζεται ορθά, ενώ όχι στην πρώτη. Ως καλύτερη κρίνεται η δεύτερη ED2=49,2, χωρίς ωστόσο τόσο μεγάλες διαφορές με την άλλη εφαρμογή.



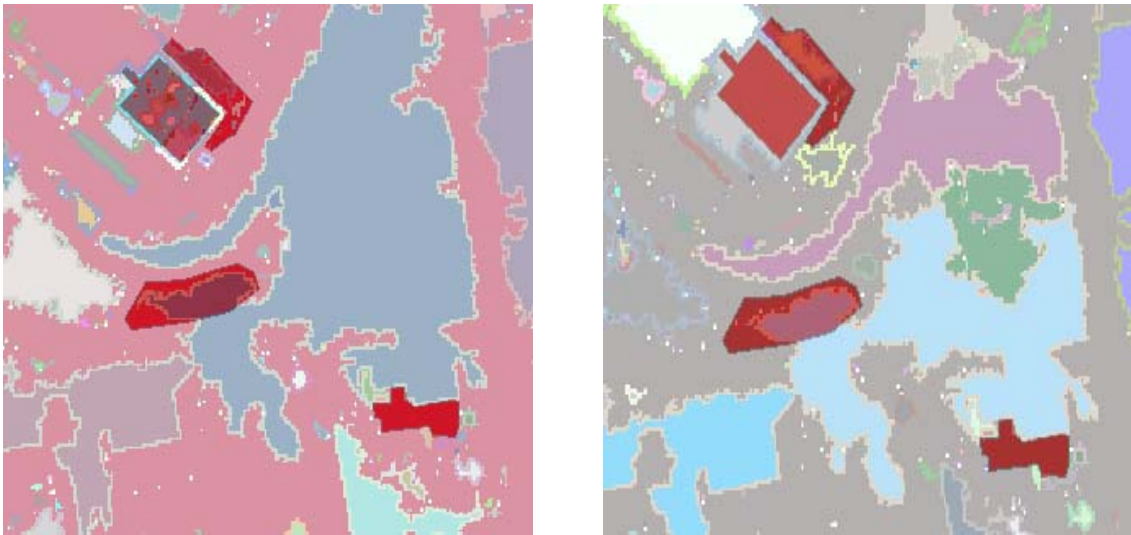
Εικόνα 3.68 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED2 (Region Growing - Min Fitness / Min Raw = 131/42,86)



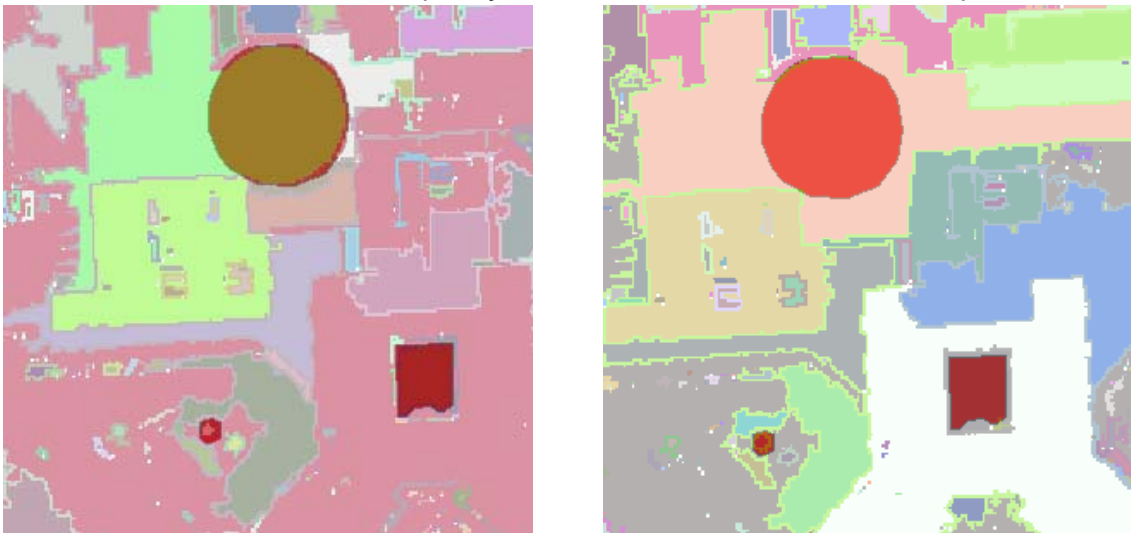
Εικόνα 3.69 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED2 (Region Growing - Min Fitness / Min Raw = 42,1 / 49,2)



Εικόνα 3.70 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 131 / 42,86) - δεξιά Min Fitness / Min Raw = 42,1 / 49,2)



Εικόνα 3.71 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 131 / 42,86) - δεξιά Min Fitness / Min Raw = 42,1 / 49,2)

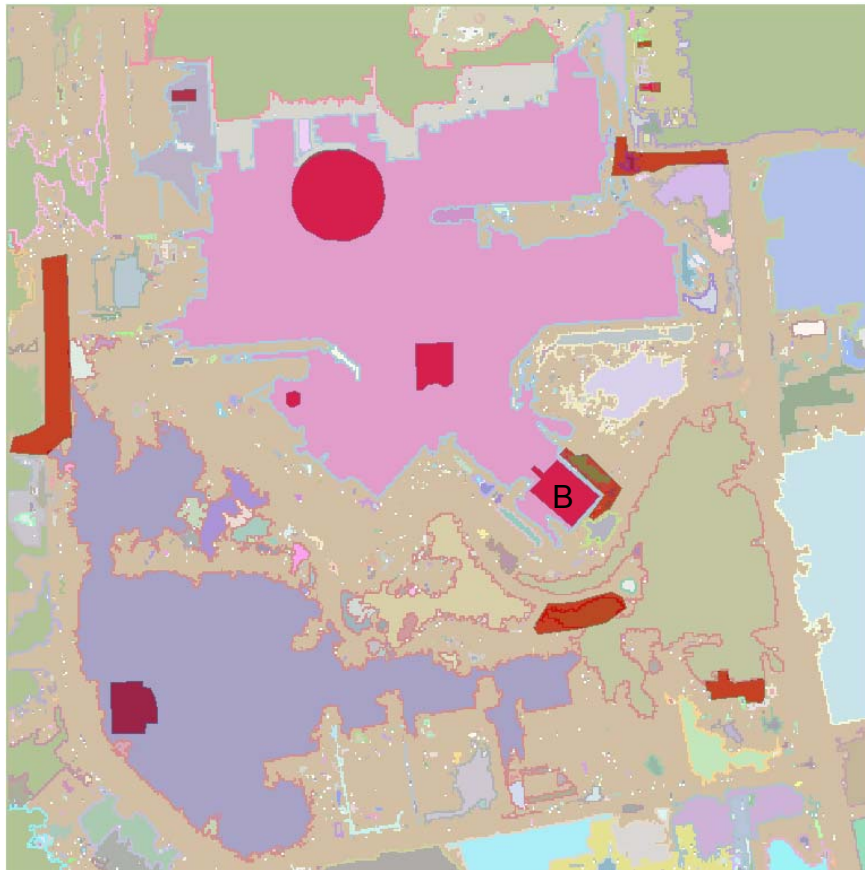


Εικόνα 3.72 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης (αριστερά Min Fitness / Min Raw = 131 / 42,86) - δεξιά Min Fitness / Min Raw = 42,1 / 49,2)

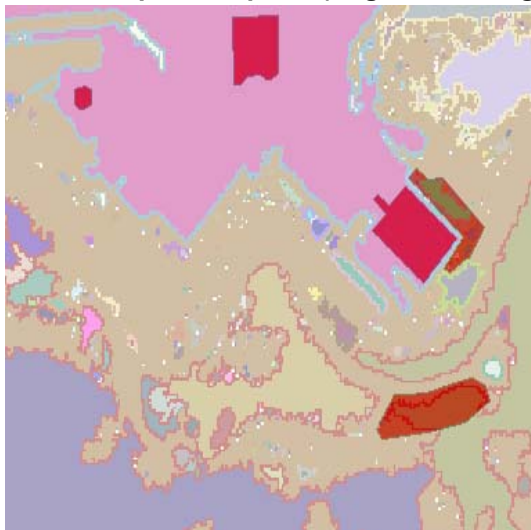
Συνάρτηση αξιολόγησης ED1:

Η χρησιμοποίηση του δείκτη ED1 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED1=0,70$ με τιμές παραμέτρων $P1=53$, $P2=31$ και $P3=147$ και όπως διαπιστώθηκε στην χρήση του δείκτη ED2 η τιμή αυτή είναι η βέλτιστη. Για τις τιμές αυτές ο δείκτης ED2 λαμβάνει τιμή $ED2=51,089$, λίγο μεγαλύτερη από τις αντίστοιχες βέλτιστες με χρήση του δείκτη ED2. Όπως παρατηρήθηκε και πριν για την συγκεκριμένη εικόνα με τον συγκεκριμένο αλγόριθμο οι δυο δείκτες έδρασαν σχεδόν το ίδιο.

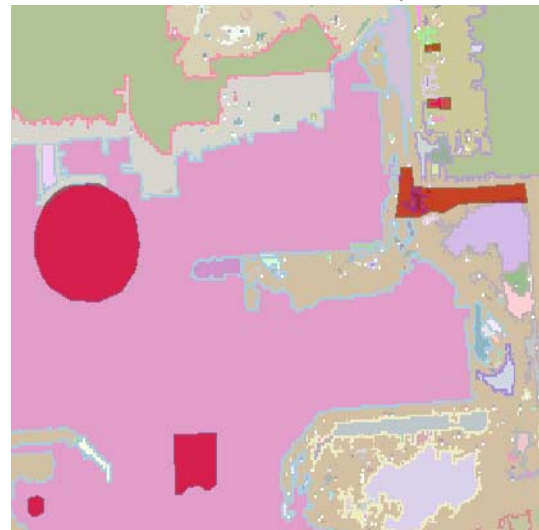
Από την κατάτμηση δημιουργούνται συνολικά 1809 αντικείμενα. Το μέγεθος αυτό είναι σχεδόν ίδιο με το πλήθος των αντικειμένων με χρήση του δείκτη ED2. Όπως παρουσιάζεται και στις εικόνες 3.73 έως 3.75 η κατάτμηση δεν μπορεί να χαρακτηριστεί ως ορθή, όπως και στις εικόνες 3.68 και 3.69, λαμβάνοντας υπόψη τα αντικείμενα αναφοράς. Ωστόσο παρατηρείται ότι υπάρχει το αντικείμενο «B», οποίο έχει εντοπιστεί ορθά όπως και στην δεύτερη περίπτωση με χρήση του δείκτη ED2. Ο ΓΑ ελαχιστοποιώντας τον δείκτη ED1, λειτούργησε μεν ορθά όμως η ελαχιστοποίηση του δείκτη δεν αποδίδει τα βέλτιστα αποτελέσματα, καθώς δημιουργείται πλήθος αντικειμένων.



Εικόνα 3.73 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED1 (Region Growing - Min Fitness / Min Raw = 0,61 / 0,70)



Εικόνα 3.74 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



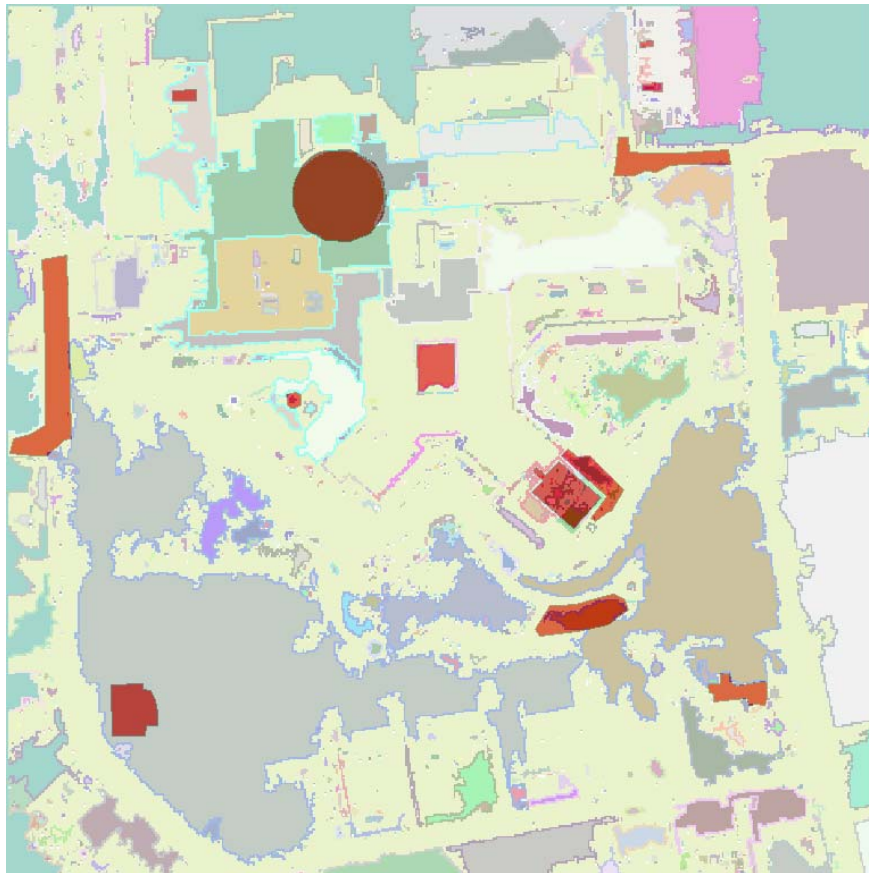
Εικόνα 3.75 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

Συνάρτηση αξιολόγησης $(ED1+ED2)/2$:

Η χρησιμοποίηση του δείκτη $(ED1+ED2)/2$ ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή 21. Συγκεκριμένα τιμές $ED1=0,701$ και $ED2=42,989$ με τιμές παραμέτρων $P1=147$, $P2=172$ και $P3=50$. Παρατηρείται ότι με την χρησιμοποίηση του δείκτη αυτού ο ΓΑ καταλήγει σχεδόν στις βέλτιστες τιμές των δυο δεικτών που παρατηρήθηκαν στις προηγούμενες εφαρμογές. Οι τιμές των δεικτών είναι δηλαδή οι ελάχιστες που

παρατηρήθηκαν στις προηγούμενες εφαρμογές. Το γεγονός αυτό αποδεικνύει την ορθή λειτουργία του ΓΑ όπου προσπαθεί να ελαχιστοποιήσει και τις δυο τιμές ταυτόχρονα.

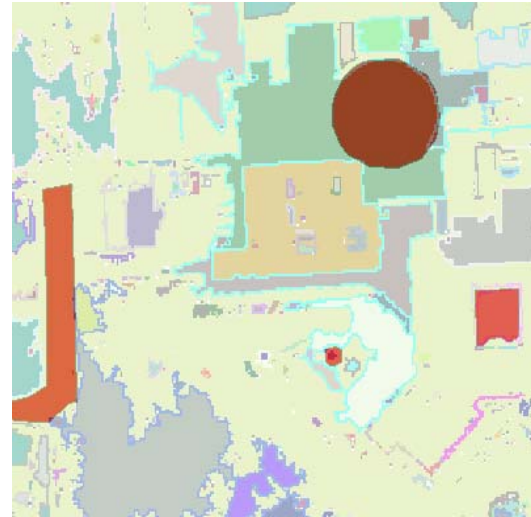
Από την κατάτμηση δημιουργούνται συνολικά 1988 αντικείμενα. Το πλήθος αυτό είναι σχεδόν το ίδιο με απόκλιση περίπου γύρω στα 90 αντικείμενα με το πλήθος που προέκυψε στις εφαρμογές του ΓΑ με χρήση των δεικτών ED1 και ED2 ξεχωριστά, γεγονός αναμενόμενο όπως αναφέρθηκε ήδη λόγω των τιμών των δεικτών. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση δεν μπορεί να χαρακτηριστεί ως η βέλτιστη δυνατή. Στις εικόνες 3.76 έως 3.78 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με τα αντικείμενα αναφοράς. Στην εικόνα 3.76 παρουσιάζεται το σύνολο της εικόνας ενώ στις υπόλοιπες τμηματικά τα αντικείμενα αναφοράς για καλύτερη εποπτική παρουσίαση των αποτελεσμάτων. Γενικότερα παρατηρείται ότι η κατάτμηση ακολουθεί κυρίως την κατάτμηση της εικόνας 3.68.



Εικόνα 3.76 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη $(ED1+ED2)/2$ (Region Growing - Min Fitness / Min Raw = 22 / 21)



Εικόνα 3.77 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.78 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

3.5.3.3 Υλοποίηση στην ΔΕ της NASA για τον πλανήτη Άρη

Τα αποτελέσματα των κατατμήσεων που προέκυψαν χρησιμοποιώντας τον συγκεκριμένο αλγόριθμο κατάτμησης στην υλοποίηση του ΓΑ, εξαρτώνται από την συνάρτηση αξιολόγησης που χρησιμοποιείται κάθε φορά. Καλύτερα αποτελέσματα παρατηρούνται με την χρησιμοποίηση του δείκτη ED1, σε σχέση με τους υπόλοιπους δυο, ως προς το αποτέλεσμα της κατάτμησης, γεγονός αντίθετο από τα μέχρι τώρα διαπιστωθέντα. Ακολουθεί ο δείκτης ED2 και ο $(ED1+ED2)/2$.

Όσον αφορά τον χρόνο εκτέλεσης του ΓΑ αυτός κυμαίνεται από 43 ώρες έως 93 ώρες ανάλογα με τις επιλογές του ΓΑ και κυρίως με το πλήθος γενεών και τον εκάστοτε πληθυσμό. Για παράδειγμα με χρήση του δείκτη ED2 ως συνάρτηση αξιολόγησης καλύτερα αποτελέσματα προέκυψαν σε 75 ώρες στην πρώτη περίπτωση του πίνακα 3.6. Η διαφορά μέγιστου και ελάχιστου χρόνου οφείλεται στις διαφορετικές διαδικασίες υλοποίησης του ΓΑ σε κάθε εφαρμογή του. Ιδίως το πλήθος του πληθυσμού και των γενεών είναι οι καθοριστικοί παράγοντες του χρόνου εκτέλεσης του ΓΑ. Το πεδίο έρευνας του ΓΑ δεν πρέπει ούτε να περιορίζεται αλλά ούτε και να είναι εντελώς διευρυμένο γιατί στην μεν πρώτη περίπτωση τα αποτελέσματα του ΓΑ εμπεριέχουν πολύ μεγάλο κίνδυνο να είναι εσφαλμένα ενώ στην δεύτερη περίπτωση θα δώσει κατά μεγάλη πιθανότητα ορθά αποτελέσματα αλλά σε πολύ μεγάλο χρόνο, γεγονός που μπορεί να καταστήσει τον ΓΑ μη συμφέρων.

Όσον αφορά τις επιλογές του ΓΑ στις διαδικασίες υλοποίησης του, επιλέχτηκαν σε κάθε εφαρμογή όσο το δυνατόν διαφορετικές, ώστε να ελεγχθούν τα αποτελέσματα που προκύπτουν. Με την χρήση του συγκεκριμένου αλγόριθμου, καθώς το εύρος τιμών των παραμέτρων ήταν το ίδιο, δόθηκε η δυνατότητα για χρήση περισσότερων επιλογών στην διαδικασία μετάλλαξης του ΓΑ. Ωστόσο στην διαδικασία της αρχικοποίησης εφαρμόστηκε η μέθοδος «Integer» κυρίως, ώστε να δημιουργούνται άτομα με ακέραιες τιμές.

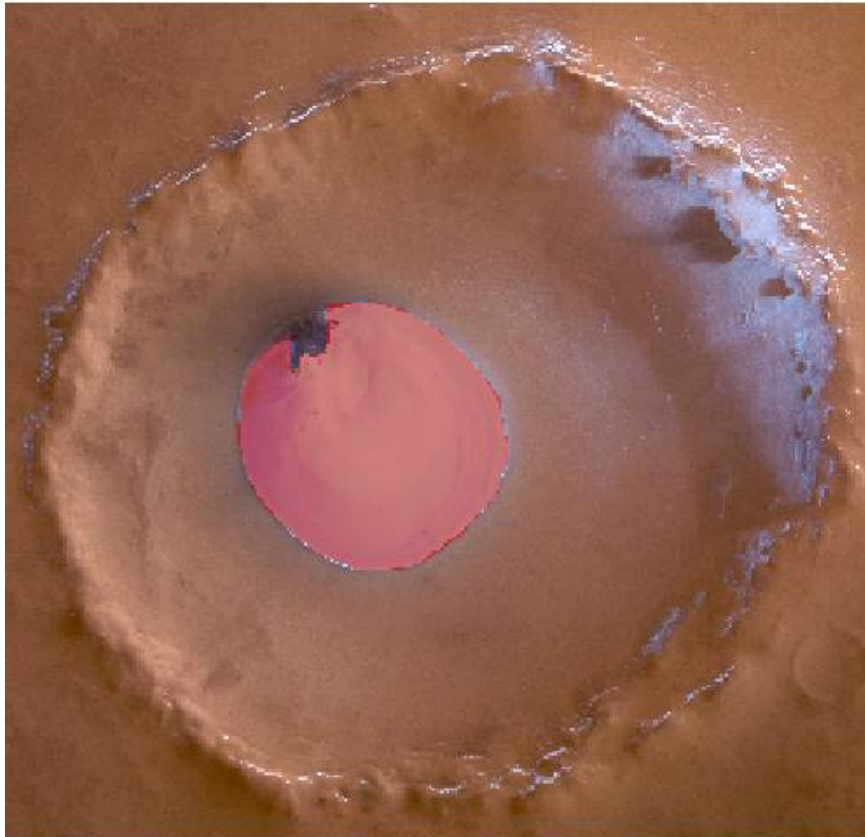
Γενικότερα παρατηρείται ότι για την συγκεκριμένη εικόνα τα αποτελέσματα της κατάτμησης κρίνονται βέλτιστα για τον δείκτη ED1 και ικανοποιητικά για τους άλλους δυο. Εντοπίζεται με ορθότητα και πληρότητα σε ποσοστό σχεδόν 100% το αντικείμενο αναφοράς. Για στατιστικούς λόγους αναφέρεται ότι ο συνολικός χρόνος εκτέλεσης των πέντε εφαρμογών ανέρχεται σε περίπου 349 ώρες με μέσο όρο τις 70 ώρες.

ΔΕ NASA για τον πλανήτη Άρη 299pixel x 307pixel

Ημερ Έναρξης	Χρόνος Εκτ/σης	Pop Size	Initiali- zation	Selection	Crossover	Mutation	Gen	mutation rate	crossover rate	LB (P1_P2_P3)	UB (P1_P2_P3)	Συνάρτηση Αξιολόγησης	ΑΠΟΤΕΛΕΣΜΑΤΑ				
													P1	P2	P3	conn	Min Fitness /Min Raw
26/4/13	75h	50	Integer	Rank	Edge	Integer Gaussian	40	2%	80%	20	230	ED2	211	158	128	8	1,27/ 0,158
29/4/13	43h	10	Allele	Tourna- ment	Edge	SIM	100	2%	80%	10_30_30	220_200_200	ED2	198	173	153	4	4,54/0,158
30/4/13	93h	10	Integer	Uniform	Two Point	Integer Gaussian	200	3%	90%	20	220	ED2	99	166	138	8	3,44/ 0,158
10/5/13	60h	10	Integer	Tourna- ment	Two Point	Integer Gaussian	150	2%	80%	30	210	ED1	71	118	112	8	0,20/0,02
17/5/13	78,5h	50	Integer	Tourna- ment	Edge	SIM	10	2%	85%	30	230	(ED1+ED2)/2	174	170	145	8	0,91/ 0,13

Πίνακας 3.6 - Στοιχεία εφαρμογής ΓΑ στην ΔΕ της NASA για τον πλανήτη Άρη, με αλγόριθμο κατάτμησης region growing

Οι βέλτιστες κατατμήσεις που προκύπτουν για την εικόνα, για κάθε μια από τις τρεις συνάρτησης αξιολόγησης, με βάση τα αποτελέσματα του πίνακα 3.6 παρουσιάζονται στις εικόνες παρακάτω. Στην εικόνα 3.79 παρουσιάζονται το αντικείμενο αναφοράς σε επικάλυψη με την εικόνα από τον αερομεταφερόμενο σαρωτή, για την εποπτική παρουσίαση αυτού.

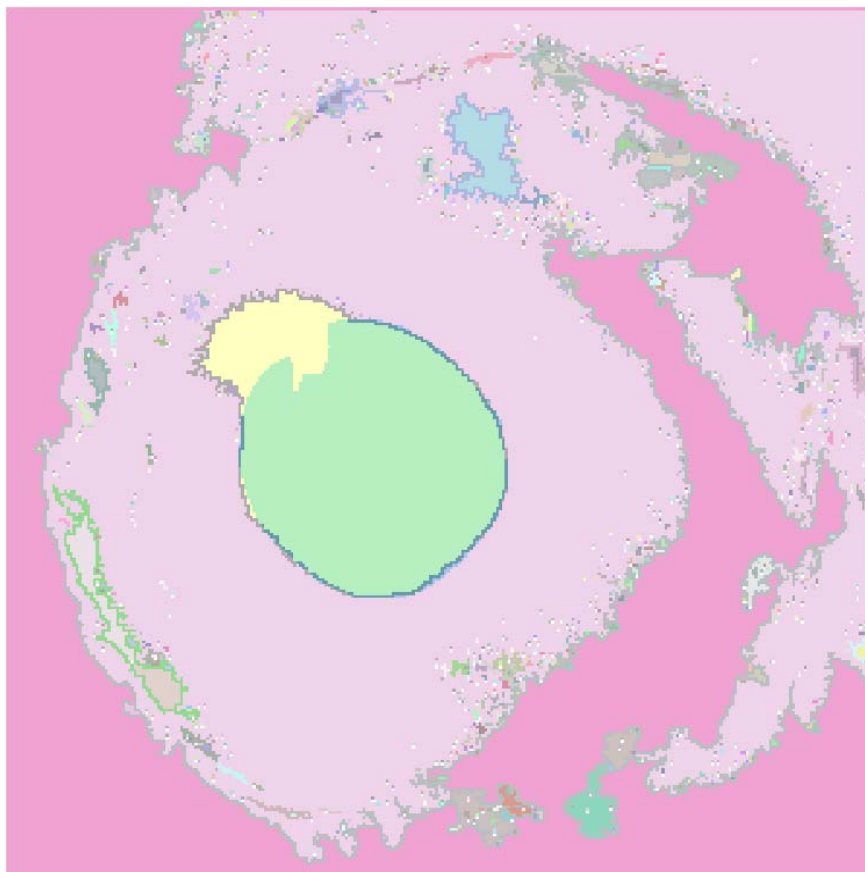


Εικόνα 3.79 - Αντικείμενο αναφοράς σε επικάλυψη με την εικόνα του πλανήτη Άρη

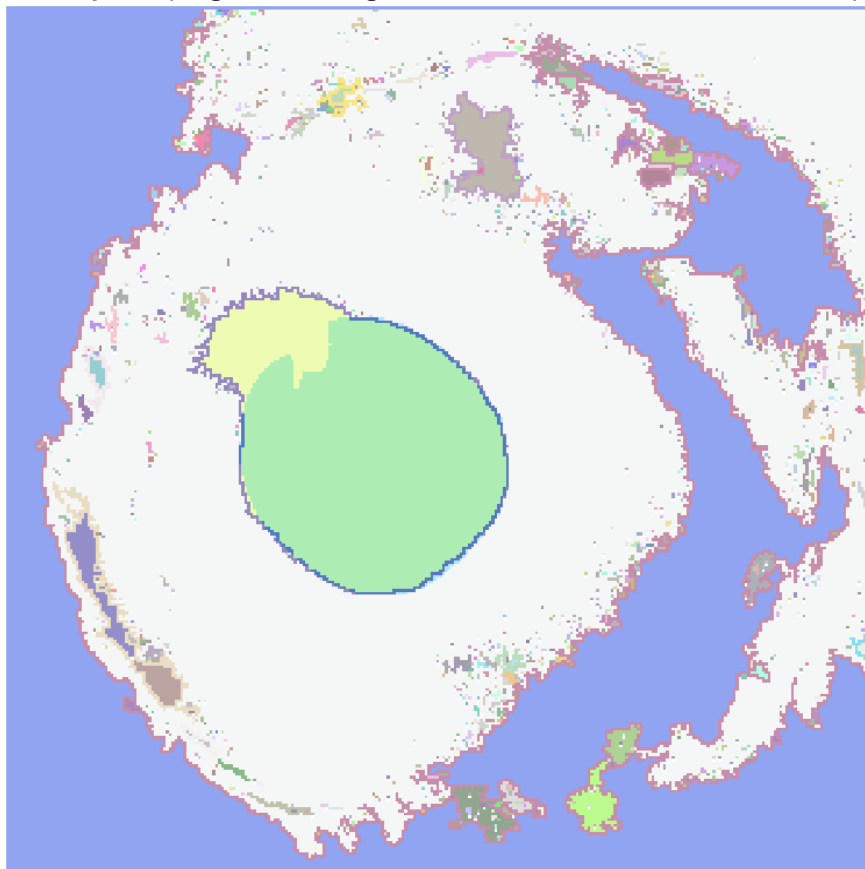
Συνάρτηση αξιολόγησης ED2:

Η χρησιμοποίηση του δείκτη ED2 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED2=0,158$ και στις τρεις εφαρμογές του ΓΑ. Ωστόσο η τιμή καταλληλότητας ήταν διαφορετική σε κάθε μια. Η βέλτιστη από πλευράς τιμής καταλληλότητας ήταν η πρώτη με τιμή 1,27. Για την εφαρμογή αυτή οι τιμές παραμέτρων ήταν $P1=211$, $P2=158$ και $P3=128$ ενώ για τιμή καταλληλότητας 3,44 οι τιμές των παραμέτρων ήταν $P1=99$, $P2=166$ και $P3=138$. Ο δείκτης ED1 λαμβάνει τιμή 0.104 στην πρώτη περίπτωση και τιμή 0,105 στην δεύτερη περίπτωση. Παρατηρείται ότι η τιμή αυτή δεν είναι η βέλτιστη με την χρησιμοποίηση του ED1 ως δείκτη για την συνάρτηση αξιολόγησης, καθώς η βέλτιστη τιμή είναι η $ED1=0,02$.

Από την πρώτη κατάτμηση δημιουργούνται συνολικά 1376 αντικείμενα ενώ από την δεύτερη 1374. Παρατηρείται ότι και στις δυο περιπτώσεις το πλήθος των αντικειμένων είναι υπερβολικά μεγάλο. Ωστόσο το αντικείμενο αναφοράς εντοπίστηκε ορθά σε μεγάλο ποσοστό και στις δυο περιπτώσεις όπως παρουσιάζεται στις εικόνες 3.80 και 3.81, σε επικάλυψη με το αντικείμενο αναφοράς. Το γεγονός της δημιουργίας πλήθους αντικειμένων ωστόσο οδηγεί στο συμπέρασμα ότι η χρήση του δείκτη ED2 σε εικόνες όπου υπάρχει ένα αντικείμενο δεν λειτουργεί ορθά στην αποτελεσματικότητα της κατάτμησης, με χρήση του συγκεκριμένου αλγορίθμου κατάτμησης. Η συγκεκριμένη κατάτμηση κατ' επέκταση δεν μπορεί να χαρακτηριστεί ως η βέλτιστη δυνατή αν και η χρήση του ΓΑ απέδωσε ικανοποιητικά αποτελέσματα τείνοντας να μηδενίσει την τιμή της συνάρτησης αξιολόγησης.



Εικόνα 3.80 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (Region Growing - Min Fitness / Min Raw = 1,27 / 0,158)

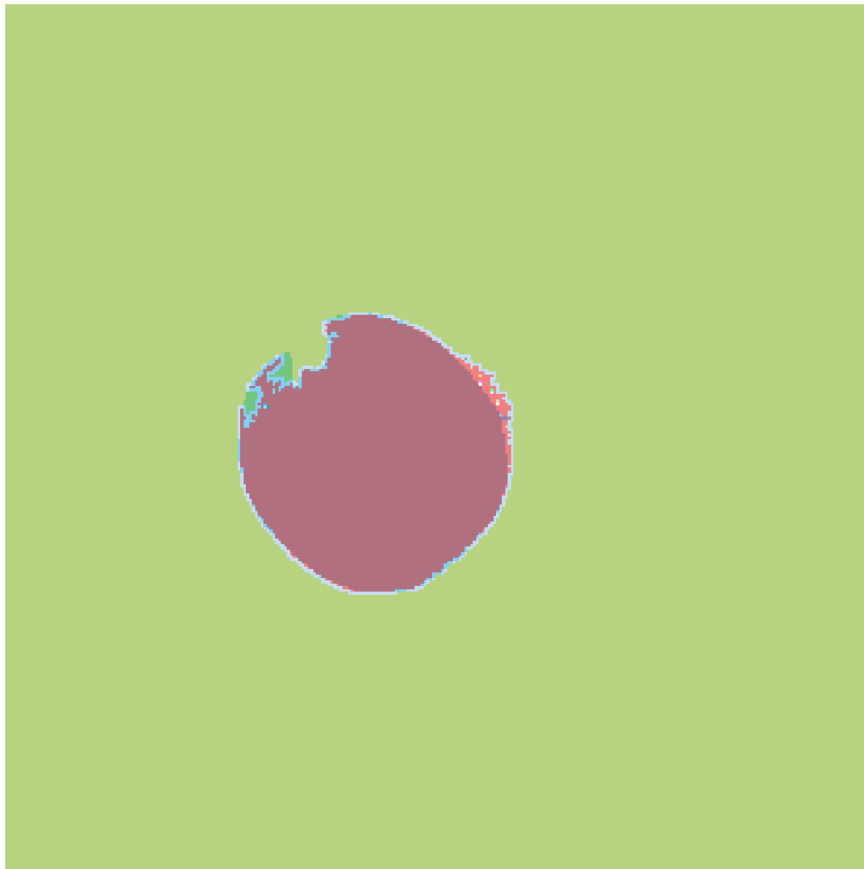


Εικόνα 3.81 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (Region Growing - Min Fitness / Min Raw = 3,44 / 0,158)

Συνάρτηση αξιολόγησης ED1:

Η χρησιμοποίηση του δείκτη ED1 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED1=0,02$ με τιμές παραμέτρων $P1=71$, $P2=118$ και $P3=112$. Για τις τιμές αυτές ο δείκτης ED2 λαμβάνει τιμή $ED2=5$. Η ελάχιστη τιμή του δείκτη ED2, όπως ήδη αναφέρθηκε είναι 0,158, χωρίς ωστόσο να προκύπτει βέλτιστη κατάτμηση σε αυτή.

Από την κατάτμηση δημιουργούνται συνολικά 15 αντικείμενα. Το μέγεθος αυτό είναι βέλτιστο και ιδίως αν παρατηρήσει κανείς την εικόνα 3.82 όπου το αντικείμενο αναφοράς έχει εντοπιστεί πλήρως, ενώ η υπόλοιπη εικόνα εμφανίζεται σαν ένα αντικείμενο. Τα υπόλοιπα 13 αντικείμενα είναι ελάχιστης έκτασης, τα οποία θα μπορούσαν να εξαλειφθούν με χρήση κάποιου φίλτρου. Παρατηρείται ότι η κατάτμηση είναι βέλτιστη και στην περίπτωση αυτή η τιμή του δείκτη ED2 είναι 5, γεγονός που αποδεικνύει για άλλη μια φορά την μη ορθή λειτουργία του δείκτη σε περίπτωση ύπαρξης ενός αντικειμένου αναφοράς. Αντίθετα η χρήση του δείκτη ED1 απέδωσε πλήρως ικανοποιητικά αποτελέσματα.

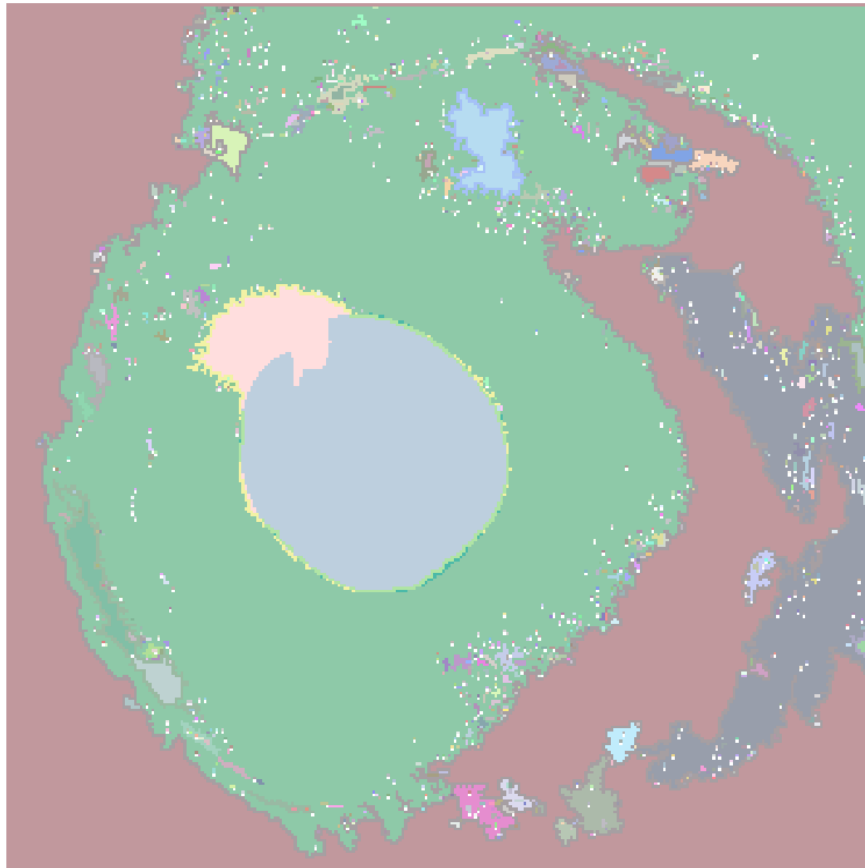


Εικόνα 3.82 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED1 (Region Growing - Min Fitness / Min Raw = 0,20 / 0,02)

Συνάρτηση αξιολόγησης $(ED1+ED2)/2$:

Η χρησιμοποίηση του δείκτη $(ED1+ED)/2$ ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή 0,13. Συγκεκριμένα τιμές $ED1=0,104$ και $ED2=0,158$, δηλαδή την ελάχιστη που εντοπίστηκε με χρήση μόνο του δείκτη ED2, με τιμές παραμέτρων $P1=174$, $P2=170$ και $P3=145$. Παρατηρείται ότι με την χρησιμοποίηση του δείκτη αυτού η προκύπτουσα κατάτμηση ακολουθεί κυρίως την κατάτμηση που προκύπτει με την χρήση του δείκτη ED2.

Από την κατάτμηση δημιουργούνται συνολικά 1378 αντικείμενα. Το πλήθος αυτό είναι το σχεδόν ίδιο με το πλήθος των αντικειμένων με χρησιμοποίηση του δείκτη ED2 ως συνάρτηση αξιολόγησης. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση δεν μπορεί να χαρακτηριστεί ως η βέλτιστη δυνατή. Στην εικόνα 3.83 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με το αντικείμενο αναφοράς.



Εικόνα 3.83 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη $(ED1+ED2)/2$ (Region Growing - Min Fitness / Min Raw = 0,91 / 0,13)

3.5.4 Υλοποίηση Γενετικού Αλγορίθμου, με Χρησιμοποίηση του Αλγορίθμου Κατάτμησης Region Merging

Στην ενότητα αυτή πραγματοποιείται εφαρμογή του ΓΑ με χρησιμοποίηση του αλγορίθμου κατάτμησης Region Merging. Αρχικά υλοποιήθηκαν πλήθος από δοκιμές στην εικόνα 3.2 (Calw) που ωστόσο λόγω του πειραματικού τους χαρακτήρα αλλά και των μη αναμενόμενων, πολλές φορές, αποτελεσμάτων λόγω διερεύνησης της διαδικασίας δεν αναφέρονται στα εδάφια που ακολουθούν. Στα εδάφια 3.5.4.1, 3.5.4.2 και 3.5.4.3 παρουσιάζονται τα αποτελέσματα για κάθε μια από τις τρεις εικόνες που χρησιμοποιούνται στην διαδικασία υλοποίησης του ΓΑ.

3.5.4.1 Υλοποίηση στην ΔΕ του Αγίου Στεφάνου Αττικής

Τα αποτελέσματα των βέλτιστων κατατμήσεων που προέκυψαν χρησιμοποιώντας τον συγκεκριμένο αλγόριθμο κατάτμησης στην υλοποίηση του ΓΑ, παρατηρείται ότι δεν εξαρτώνται, σχεδόν καθόλου, από την συνάρτηση αξιολόγησης που χρησιμοποιείται κάθε φορά. Τα αποτελέσματα που προέκυψαν είναι σχεδόν τα ίδια, με πολύ μικρές διαφορές στις κατατμήσεις.

Όσον αφορά τον χρόνο εκτέλεσης του ΓΑ αυτός κυμαίνεται από 38 ώρες, όπου προκύπτει και η βέλτιστη κατάτμηση για τον δείκτη ED2, έως 98 ώρες ανάλογα με τις επιλογές του ΓΑ και κυρίως με το πλήθος γενεών και τον εκάστοτε πληθυσμό. Στην τρίτη εφαρμογή του πίνακα 3.7 με χρήση του δείκτη ED2, παρατηρείται ότι ο χρόνος υλοποίησης είναι σχεδόν διπλάσιος από την εφαρμογή ένα καθώς διπλασιάστηκε το μέγεθος του πληθυσμού. Ωστόσο το αποτέλεσμα της κατάτμησης δεν ήταν καλύτερο παρά την καθυστέρηση μεγάλη χρόνου στην εφαρμογή του ΓΑ.

Όσον αφορά τις επιλογές του ΓΑ στις διαδικασίες υλοποίησης του, επιλέχτηκαν σε κάθε εφαρμογή όσο το δυνατόν διαφορετικές, ώστε να ελεγχθούν τα αποτελέσματα που προκύπτουν. Με την χρήση του συγκεκριμένου αλγορίθμου, καθώς το εύρος τιμών των παραμέτρων ήταν το ίδιο, δόθηκε η δυνατότητα για χρήση περισσότερων επιλογών στην διαδικασία μετάλλαξης του ΓΑ. Ωστόσο στην διαδικασία της αρχικοποίησης εφαρμόστηκε η μέθοδος «Integer», ώστε να δημιουργούνται άτομα με ακέραιες τιμές.

Γενικότερα παρατηρείται ότι για την συγκεκριμένη εικόνα τα αποτελέσματα της κατάτμησης δεν είναι τα βέλτιστα δυνατά για κανένα δείκτη, ωστόσο μπορούν να χαρακτηριστούν ως ικανοποιητικά. Εντοπίζεται με ορθότητα και πληρότητα ένα αντικείμενο από τα εννέα χρησιμοποιώντας και τους τρεις δείκτες, ενώ οι διαφοροποιήσεις από την χρήση αυτών είναι μηδαμινές. Επιπλέον παρατηρείται ότι ο χρόνος εκτέλεσης του ΓΑ, όπως και στον προηγούμενο αλγόριθμο, εξαρτάται από τις επιλογές στις διαδικασίες του ΓΑ. Ιδίως το πλήθος του πληθυσμού και των γενεών είναι οι καθοριστικοί παράγοντες του χρόνου εκτέλεσης του ΓΑ. Για τον λόγο αυτό πρέπει αυτοί να επιλέγονται με ιδιαίτερη προσοχή ώστε να επιτυγχάνονται τα βέλτιστα αποτελέσματα στον συντομότερο χρόνο.

Για στατιστικούς λόγους αναφέρεται ότι ο συνολικός χρόνος εκτέλεσης των επτά εφαρμογών ανέρχεται σε περίπου 399 ώρες με μέσο όρο τις 57 ώρες.

ΔΕ Άγιος Στέφανος 501pixel x 501pixel

Ημερ Έναρξης	Χρόνος Εκτ/σης	Pop Size	Initiali- zation	Selection	Crossover	Mutation	Gen	mutation rate	crossover rate	LB (P1_P2)	UB (P1_P2)	Συνάρτηση Αξιολόγησης	ΑΠΟΤΕΛΕΣΜΑΤΑ		
													P1	P2	Min Fitness /Min Raw
7/4/13	38h	100	Integer	Roulette Wheel	Uniform	Integer Gaussian	10	2%	90%	1	256	ED2	165	207	17,19 / 8,96
11/4/13	43h	40	Integer	Tourna- ment	Edge	SIM	30	2%	80%	1	256	ED2	156	220	14,5 / 11,65
11/4/13	98h	200	Integer	Uniform	Uniform	Integer Gaussian	10	2%	90%	1	256	ED2	148	239	21,76 / 8,96
12/4/13	47h	80	Integer	Uniform	Cut Crossfill	SIM	30	2%	85%	1	256	ED1	162	206	0,75 / 0,67
16/4/13	73h	100	Integer	Tourna- ment	Edge	SIM	20	1,5%	80%	1	250	ED1	122	244	0,28 / 0,66
11/4/13	52,5h	150	Integer	Roulette Wheel	Uniform	Integer Gaussian	10	2%	90%	1	256	(ED1+ED2)/2	240	147	9,41 / 4,81
17/4/13	48h	70	Integer	Uniform	Edge	SIM	20	3%	85%	1	500	(ED1+ED2)/2	270	428	7,73 / 4,08

Πίνακας 3.7 - Στοιχεία εφαρμογής ΓΑ στην ΔΕ ΙΚΟΝΟΣ περιοχής του Αγίου Στεφάνου Αττικής με αλγόριθμο κατάτμησης region merging

Οι βέλτιστες κατατμήσεις που προκύπτουν για την εικόνα, για κάθε μια από τις τρεις συνάρτησης αξιολόγησης, με βάση τα αποτελέσματα του πίνακα 3.7 παρουσιάζονται στις εικόνες παρακάτω. Στην εικόνα 3.84 παρουσιάζονται τα αντικείμενα αναφοράς τα οποία επικαλύπτουν την ΔΕ του Αγίου Στεφάνου, για την εποπτική παρουσίαση αυτών.



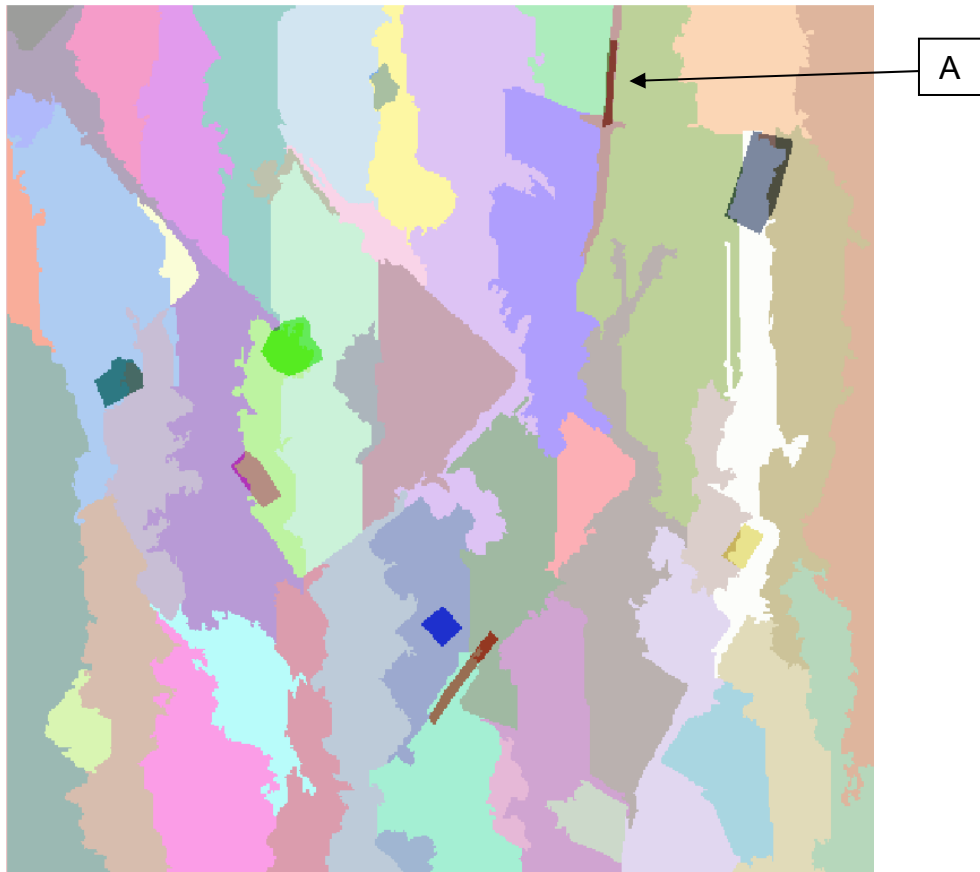
Εικόνα 3.84 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα του Αγ. Στεφάνου

Συνάρτηση αξιολόγησης ED2:

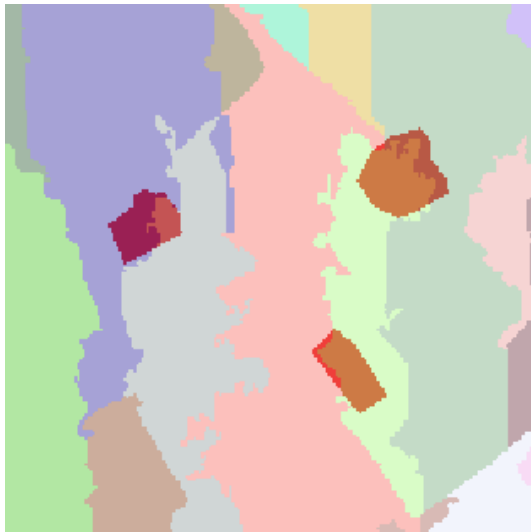
Η χρησιμοποίηση του δείκτη ED2 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED2=8,96$ με τιμές παραμέτρων $P1=165$, $P2=207$. Για τις τιμές αυτές ο δείκτης ED1 λαμβάνει τιμή $ED1=0,66$. Παρατηρείται ότι η τιμή αυτή είναι η βέλτιστη που δίνει ο ΓΑ με χρήση του δείκτη ED1 ως συνάρτηση αξιολόγησης. Κατ' επέκταση οι δυο δείκτες δίνουν τα ίδια αποτελέσματα ως προς την βέλτιστη κατάτμηση που μπορεί να επιτευχθεί με χρήση ΓΑ, γεγονός που δεν παρατηρήθηκε στον αλγόριθμο watershed.

Από την κατάτμηση δημιουργούνται συνολικά 50 αντικείμενα. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση δεν μπορεί να χαρακτηριστεί ως η βέλτιστη δυνατή ωστόσο η χρήση του ΓΑ απέδωσε ικανοποιητικά αποτελέσματα. Στις εικόνες 3.85 έως 3.87 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με τα αντικείμενα αναφοράς. Στην εικόνα 3.85 παρουσιάζεται το σύνολο της εικόνας ενώ στις υπόλοιπες τμηματικά τα αντικείμενα αναφοράς για καλύτερη εποπτική παρουσίαση των αποτελεσμάτων σε σχέση με τα αντικείμενα αναφοράς.

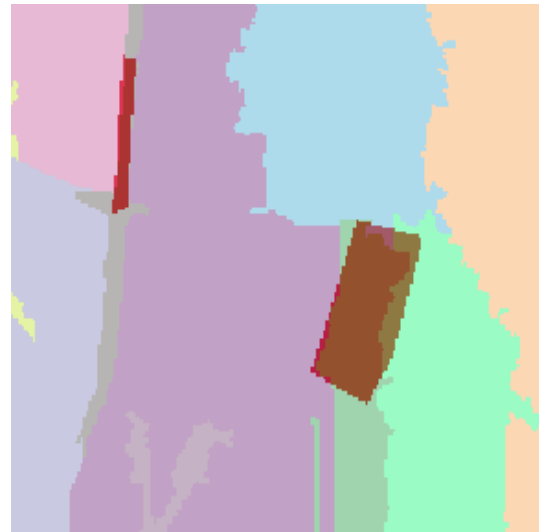
Από την κατάτμηση το αντικείμενο «Α» εντοπίστηκε ορθά, ενώ τα υπόλοιπα αντικείμενα εντοπίστηκαν ως τμήματα κυρίως μεγαλύτερων αντικειμένων και σε κάποιες περιπτώσεις ως τμήματα περισσότερων του ενός μεγαλύτερων αντικειμένων.



Εικόνα 3.85 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED2 (Region Merging - Min Fitness / Min Raw = 17,19 / 8,96)



Εικόνα 3.86 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.87 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

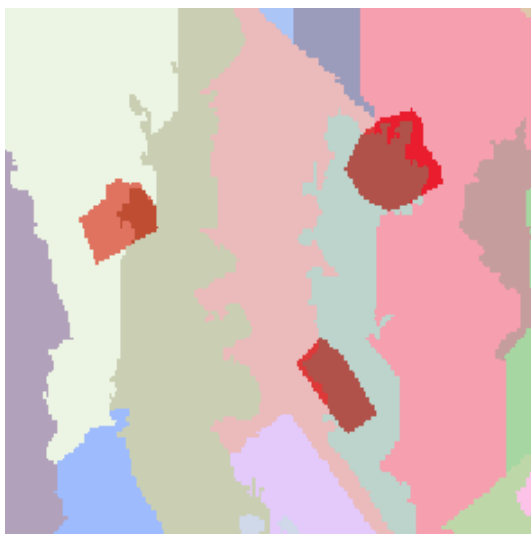
Συνάρτηση αξιολόγησης ED1:

Η χρησιμοποίηση του δείκτη ED1 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED1=0,66$ με τιμές παραμέτρων $P1=122$, $P2=244$. Επειδή ο αλγόριθμος region merging χρησιμοποιεί ως παράμετρο το άθροισμα των δυο αυτών παραμέτρων το οποίο είναι 366, παρατηρείται ότι είναι σχεδόν ίδιο με το άθροισμα με χρήση του δείκτη ED2 (372). Για τον λόγο αυτό ο δείκτης ED2 λαμβάνει σχεδόν ίδια τιμή με τον δείκτη ED1 με χρήση του ΓΑ.

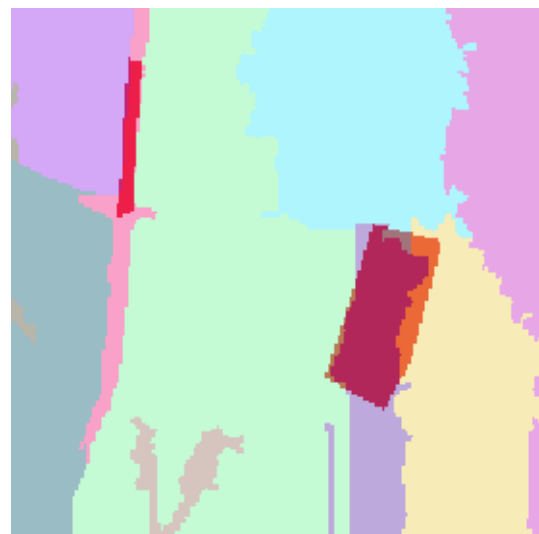
Από την κατάτμηση δημιουργούνται συνολικά 48 αντικείμενα, σχεδόν ίδια με το πλήθος των αντικειμένων με χρήση του δείκτη ED2. Όπως παρουσιάζεται και στις εικόνες 3.88 έως 3.90 η κατάτμηση που προκύπτει είναι ίδια με την κατάτμηση της εικόνας 3.85. Παρατηρείται ότι ο ED2 με τον ED1 λειτούργησαν ακριβώς με τον ίδιο τρόπο στην διαδικασία βελτιστοποίησης των παραμέτρων.



Εικόνα 3.88 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη ED1 (Region Merging - Min Fitness / Min Raw = 0,28 / 0,66)



Εικόνα 3.89 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



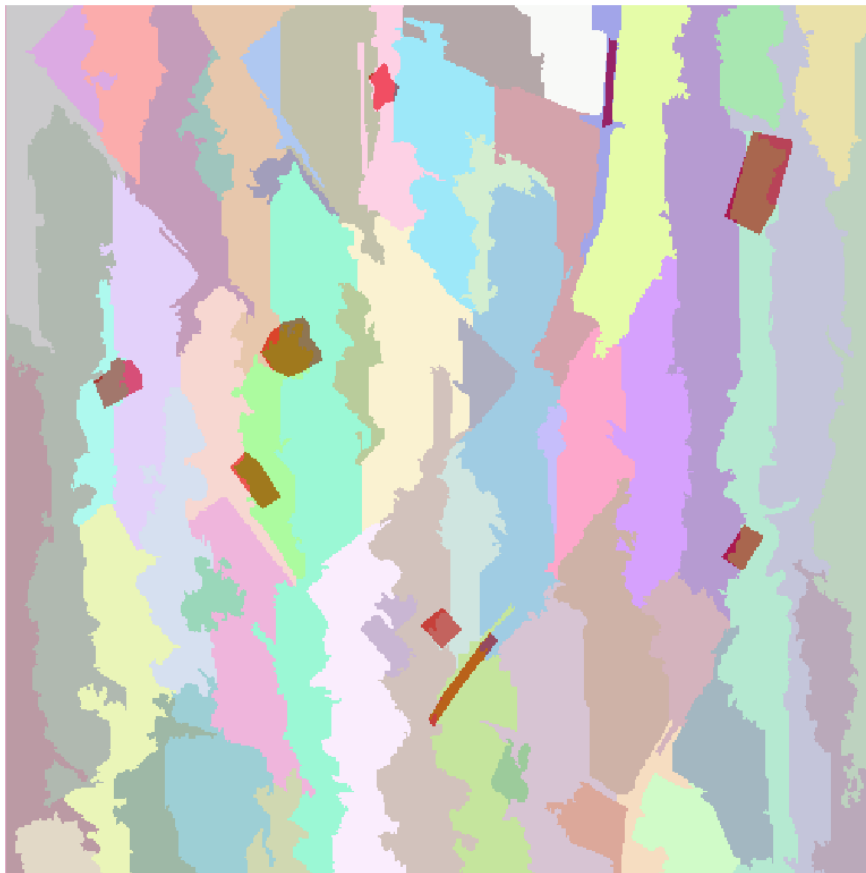
Εικόνα 3.90 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

Συνάρτηση αξιολόγησης (ED1+ED2)/2:

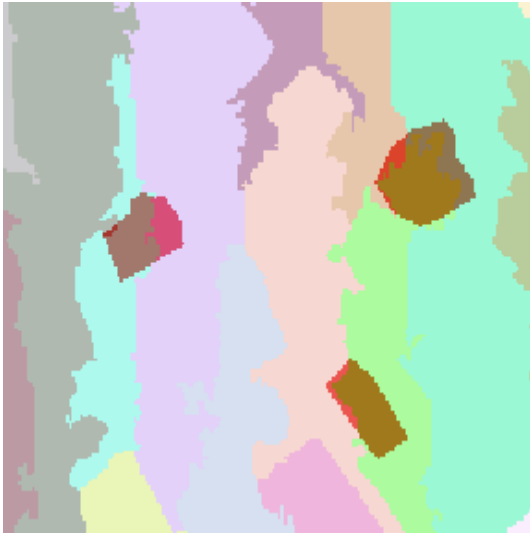
Η χρησιμοποίηση του δείκτη $(ED1+ED)/2$ ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή 4,08. Συγκεκριμένα τιμές $ED1=0,66$ και $ED2=7,52$ με τιμές παραμέτρων $P1=270$ και $P2=428$. Παρατηρείται ότι η τιμή $ED1$ είναι ίδια με αυτή που προκύπτει με την χρήση του δείκτη αυτού στο ΓΑ και η τιμή $ED2$ λίγο μικρότερη από την αντίστοιχη με χρήση του $ED2$ στον ΓΑ.

Από την κατάτμηση δημιουργούνται συνολικά 60 αντικείμενα. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση δεν μπορεί να χαρακτηριστεί ως η βέλτιστη δυνατή. Στις εικόνες 3.91 έως 3.93 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με τα αντικείμενα αναφοράς. Στην εικόνα 3.91 παρουσιάζεται το σύνολο της εικόνας ενώ στις υπόλοιπες τμηματικά τα αντικείμενα αναφοράς για καλύτερη εποπτική παρουσίαση των αποτελεσμάτων.

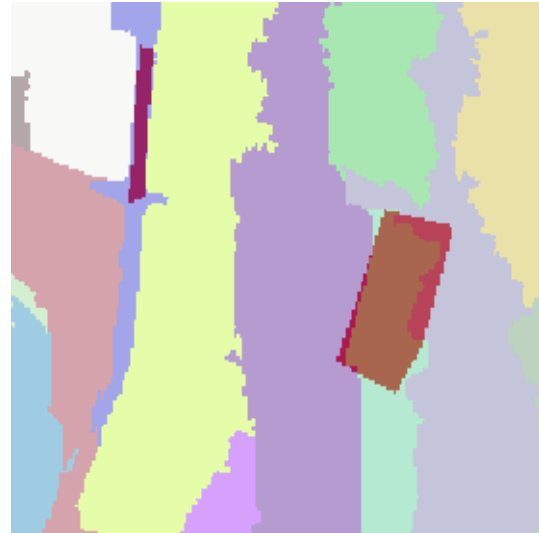
Γενικότερα παρατηρείται ότι και η κατάτμηση αυτή είναι σχεδόν ίδια με τις κατατμήσεις που προέκυψαν με χρήση των δεικτών $ED1$ και $ED2$ ως συναρτήσεις αξιολόγησης στον ΓΑ.



Εικόνα 3.91 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του Αγ. Στεφάνου, με δείκτη $(ED1+ED2)/2$ (Region Merging - Min Fitness / Min Raw = 7,73/4,08)



Εικόνα 3.92 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.93 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

3.5.4.2 Υλοποίηση στην εικόνα από αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας

Τα αποτελέσματα των βέλτιστων κατατμήσεων που προέκυψαν χρησιμοποιώντας τον συγκεκριμένο αλγόριθμο κατάτμησης στην υλοποίηση του ΓΑ, παρατηρείται ότι δεν εξαρτώνται, σχεδόν καθόλου, από την συνάρτηση αξιολόγησης που χρησιμοποιείται κάθε φορά, συμπέρασμα που προέκυψε και από την εικόνα του Αγ. Στεφάνου. Τα αποτελέσματα που προέκυψαν είναι σχεδόν τα ίδια, με πολύ μικρές διαφορές στις κατατμήσεις. Ιδίως ο δείκτης ED2 έδωσε ακριβώς τα ίδια αποτελέσματα με τον δείκτη $(ED1+ED2)/2$.

Όσον αφορά τον χρόνο εκτέλεσης του ΓΑ αυτός κυμαίνεται από 29,8 ώρες, έως 110 ώρες ανάλογα με τις επιλογές του ΓΑ και κυρίως με το πλήθος γενεών και τον εκάστοτε πληθυσμό. Η βέλτιστη κατάτμηση με χρήση του δείκτη ED2 προέκυψε στον μικρότερο χρόνο από τις τρεις εφαρμογές με τον συγκεκριμένο δείκτη. Μεγαλύτερος χρόνος εκτέλεσης του αλγορίθμου δεν συνεπάγεται και καλύτερο αποτέλεσμα πάντοτε.

Όσον αφορά τις επιλογές του ΓΑ στις διαδικασίες υλοποίησης του, επιλέχθηκαν σε κάθε εφαρμογή όσο το δυνατόν διαφορετικές, ώστε να ελεγχθούν τα αποτελέσματα που προκύπτουν. Με την χρήση του συγκεκριμένου αλγορίθμου, καθώς το εύρος τιμών των παραμέτρων ήταν το ίδιο, δόθηκε η δυνατότητα για χρήση περισσότερων επιλογών στην διαδικασία μετάλλαξης του ΓΑ. Ωστόσο στην διαδικασία της αρχικοποίησης εφαρμόστηκε η μέθοδος «Integer», ώστε να δημιουργούνται άτομα με ακέραιες τιμές.

Γενικότερα παρατηρείται ότι για την συγκεκριμένη εικόνα τα αποτελέσματα της κατάτμησης κρίνονται ικανοποιητικά και για τους τρεις δείκτες. Εντοπίζονται με ορθότητα και πληρότητα τρία αντικείμενα από τα δεκατρία, δυο εντοπίζονται σε μεγάλο ποσοστό, ενώ ικανοποιητικά είναι τα αποτελέσματα και για τα υπόλοιπα. Επιπλέον παρατηρείται ότι ο χρόνος εκτέλεσης του ΓΑ, όπως και στον προηγούμενο αλγόριθμο, εξαρτάται από τις επιλογές στις διαδικασίες του ΓΑ. Ιδίως το πλήθος του πληθυσμού και των γενεών είναι οι καθοριστικοί παράγοντες του χρόνου εκτέλεσης του ΓΑ. Για στατιστικούς λόγους αναφέρεται ότι ο συνολικός χρόνος εκτέλεσης των επτά εφαρμογών ανέρχεται σε περίπου 448 ώρες με μέσο όρο τις 64 ώρες.

Εικόνα από αερομεταφερόμενο σαρωτή (εταιρία toposys) 473pixel x 474pixel

Ημερ Έναρξης	Χρόνος Εκτ/σης	Pop Size	Initiali- zation	Selection	Crossover	Mutation	Gen	mutation rate	crossover rate	LB (P1_P2)	UB (P1_P2)	Συνάρτηση Αξιολόγησης	ΑΠΟΤΕΛΕΣΜΑΤΑ		
													P1	P2	Min Fitness /Min Raw
7/4/13	110h	60	Integer	Tourna- ment	TwoPoint	Integer Range	30	2%	80%	1	256	ED2	226	223	1,32 / 0,97
7/4/13	47h	100	Integer	Roulette Wheel	Uniform	Integer Gaussian	10	2%	90%	1	256	ED2	242	227	1,39 / 0,85
17/4/13	78h	80	Integer	Uniform	TwoPoint	Integer Range	40	2%	95%	1	256	ED2	238	239	1,44 / 0,85
08/4/13	36h	100	Integer	Roulette Wheel	Uniform	Integer Gaussian	10	2%	90%	1	256	ED1	241	228	0,40 / 0,38
09/4/13	85h	80	Integer	Uniform	Cut Crossfill	Mutator SIM	30	2%	85%	1	256	ED1	222	190	0,39 / 0,35
10/4/13	29,8h	100	Integer	Roulette Wheel	Uniform	Integer Gaussian	10	2%	90%	1	256	ED1	140	247	0,51 / 0,66
09/4/13	64h	150	Integer	Roulette Wheel	Uniform	Integer Gaussian	10	2%	90%	1	256	(ED1+ED2)/2	248	221	1,29 / 0,62

Πίνακας 3.8 - Στοιχεία εφαρμογής ΓΑ στην εικόνα από αερομεταφερόμενο σαρωτή σε περιοχή της Γερμανίας, με αλγόριθμο κατάτμησης region merging

Οι βέλτιστες κατατμήσεις που προκύπτουν για την εικόνα, για κάθε μια από τις τρεις συνάρτησης αξιολόγησης, με βάση τα αποτελέσματα του πίνακα 3.8 παρουσιάζονται στις εικόνες παρακάτω. Στην εικόνα 3.94 παρουσιάζονται τα αντικείμενα αναφοράς σε επικάλυψη με την εικόνα από τον αερομεταφερόμενο σαρωτή, για την εποπτική παρουσίαση αυτών.

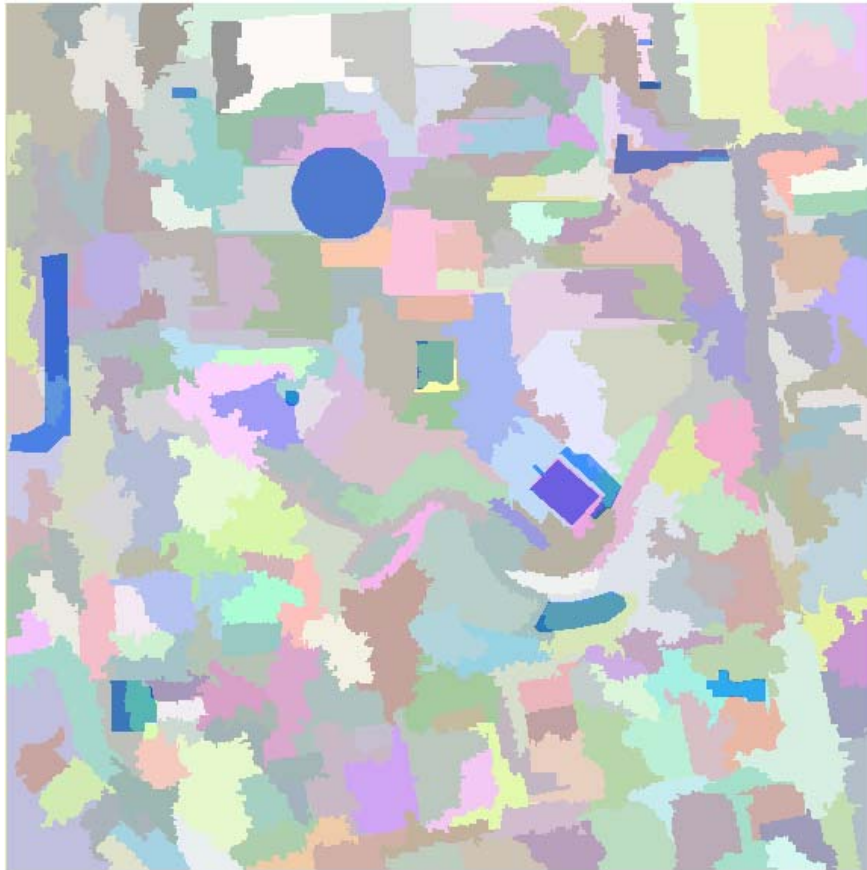


Εικόνα 3.94 - Αντικείμενα αναφοράς σε επικάλυψη με την εικόνα σε περιοχή της Γερμανίας

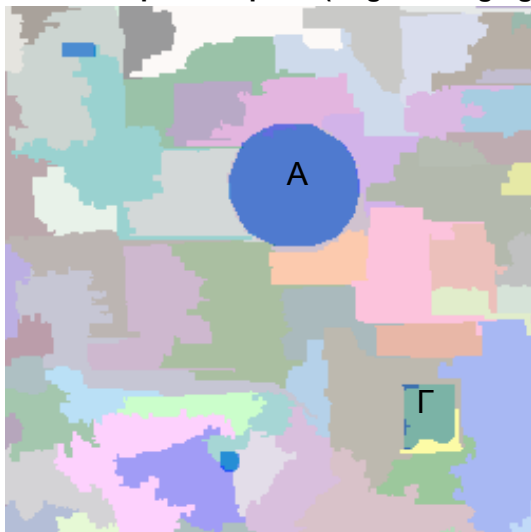
Συνάρτηση αξιολόγησης ED2:

Η χρησιμοποίηση του δείκτη ED2 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED2=0,85$ με τιμές παραμέτρων $P1=242$ και $P2=227$. Ωστόσο η τιμή καταλληλότητας δηλαδή η τιμή που προκύπτει σε σύγκριση με τις τιμές του δείκτη για την πρώτη εφαρμογή του πίνακα 3.8 είναι ελαφρά μικρότερη γι αυτό και δεν εξετάζεται η συγκεκριμένη. Για την παραπάνω τιμή ο δείκτης ED1 λαμβάνει τιμή 0,39. Η τιμή αυτή είναι κατά 0,04 μεγαλύτερη σε σχέση με την βέλτιστη τιμή που επιτεύχθηκε με τον ED1.

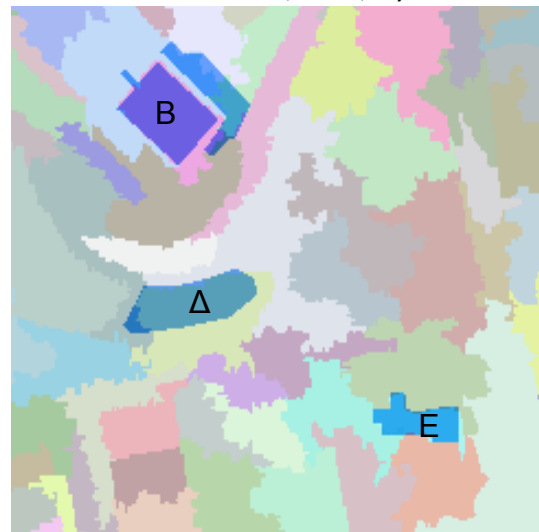
Από την κατάτμηση δημιουργούνται συνολικά 250 αντικείμενα. Λαμβάνοντας υπόψη τα αντικείμενα αναφοράς η κατάτμηση μπορεί να χαρακτηριστεί ως αρκετά ικανοποιητική. Τρία αντικείμενα έχουν εντοπιστεί πλήρως (Α, Β και Γ) και δυο αντικείμενα έχουν εντοπιστεί σε μεγάλο τμήμα τους ορθά (Δ και Ε). Η χρήση του ΓΑ απέδωσε ικανοποιητικά αποτελέσματα. Στις εικόνες 3.95 έως 3.97 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με τα αντικείμενα αναφοράς. Στην εικόνα 3.95 παρουσιάζεται το σύνολο της εικόνας ενώ στις υπόλοιπες τμηματικά τα αντικείμενα αναφοράς για καλύτερη εποπτική παρουσίαση των αποτελεσμάτων σε σχέση με τα αντικείμενα αναφοράς.



Εικόνα 3.95 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED2 (Region Merging - Min Fitness / Min Raw = 1,39 / 0,85)



Εικόνα 3.96 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



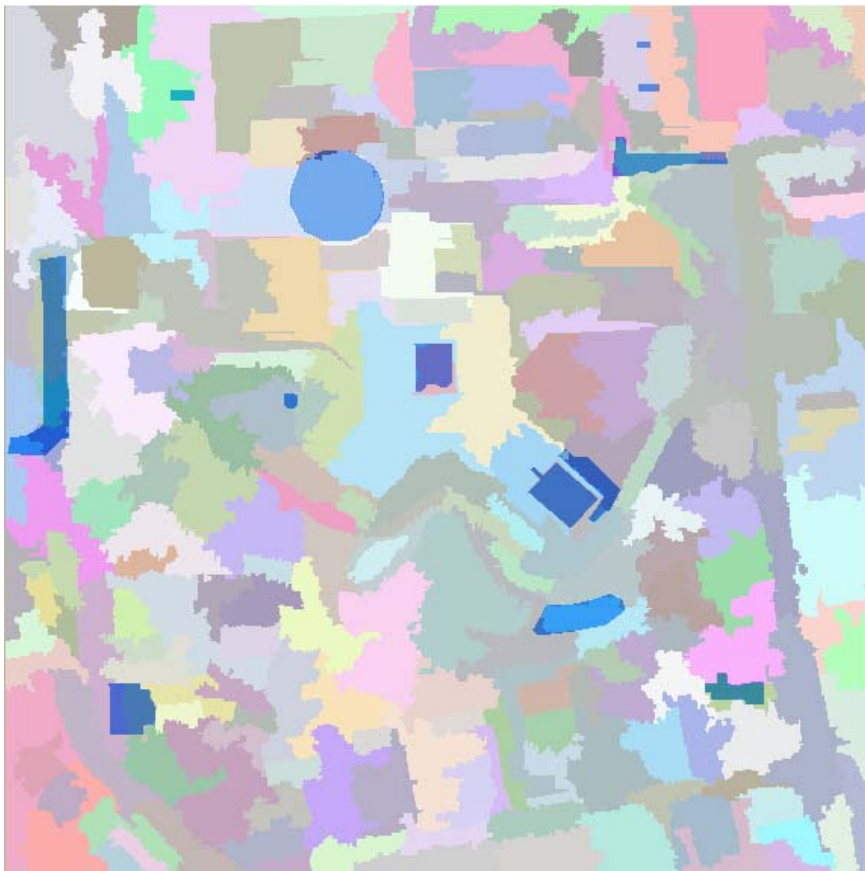
Εικόνα 3.97 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης

Συνάρτηση αξιολόγησης ED1:

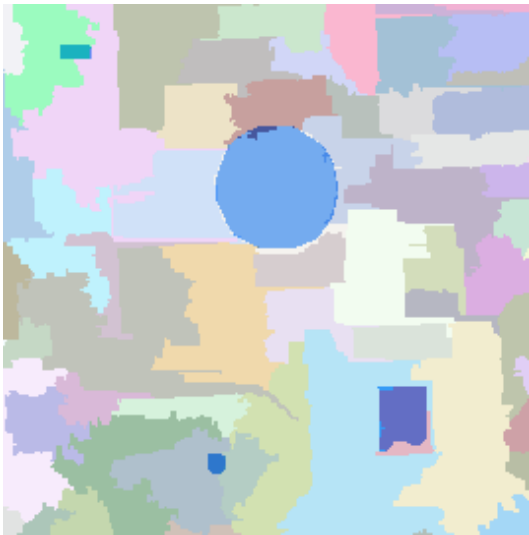
Η χρησιμοποίηση του δείκτη ED1 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED1=0,35$ με τιμές παραμέτρων $P1=222$ και $P2=190$. Το άθροισμα των δυο παραμέτρων είναι 412, ενώ το άθροισμα των παραμέτρων για την εφαρμογή του δείκτη ED2 είναι 469. Παρατηρείται μια μικρή διαφορά ή οποία είναι ωστόσο μεγαλύτερη σε σχέση με την εικόνα του Αγίου Στεφάνου όπου η διαφορά ήταν μόνο 6 μονάδες. Η τιμή που λαμβάνει ο δείκτης ED2 είναι 1,07, οπότε και προκύπτει η προαναφερθείσα διαφορά. Το γεγονός αυτό

βρίσκεται σε συμφωνία με το συμπέρασμα της προηγούμενης εικόνας ότι οι δυο δείκτες έδρασαν περίπου κατά τον ίδιο τρόπο, συγκρίνοντας το αποτέλεσμα της κατάτμησης με βάση τα αντικείμενα αναφοράς.

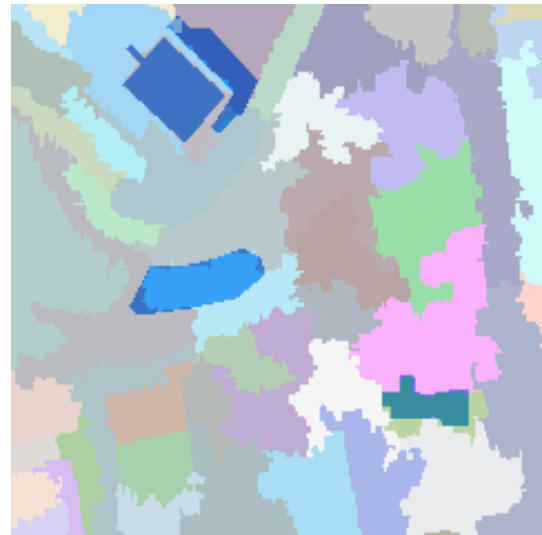
Από την κατάτμηση δημιουργούνται συνολικά 231 αντικείμενα. Το μέγεθος αυτό είναι μικρότερο από το πλήθος των αντικειμένων με την εφαρμογή του δείκτη ED2. Όπως παρουσιάζεται και στις εικόνες 3.98 έως 3.100 η κατάτμηση είναι σχεδόν η ίδια με την κατάτμηση της εικόνας 3.95 με πολύ μικρές διαφορές. Τα αντικείμενα αναφοράς που εξήχθησαν ορθά με εφαρμογή του δείκτη ED1 είναι ίδια με αυτά που εξήχθησαν με εφαρμογή του δείκτη ED2. Το συμπέρασμα αυτό προκύπτει και από τις εικόνες 3.101 και 3.102. Ο ΓΑ λειτούργησε ορθά ελαχιστοποιώντας τον δείκτη ED1, και αποδίδοντας βέλτιστο αποτέλεσμα.



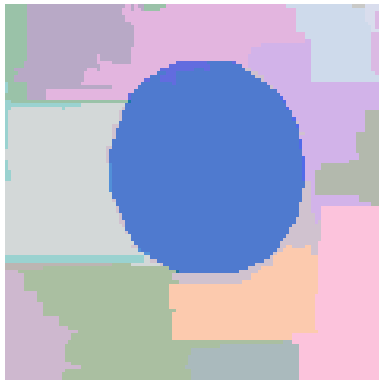
Εικόνα 3.98 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα περιοχής της Γερμανίας, με δείκτη ED1 (Region Merging - Min Fitness / Min Raw = 0,39 / 0,35)



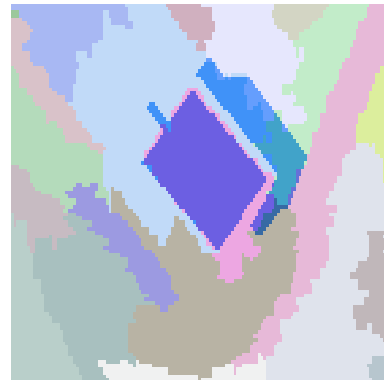
Εικόνα 3.99 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.100 - Μέρος των αντικειμένων αναφοράς σε επικάλυψη με τμήμα της εικόνας κατάτμησης



Εικόνα 3.101 - Αντικείμενα κατάτμησης με βάση την χρήση του δείκτη ED2 ως συνάρτηση αξιολόγησης (region merging)



Εικόνα 3.102 - Αντικείμενα κατάτμησης με βάση την χρήση του δείκτη ED2 ως συνάρτηση αξιολόγησης (region merging)

Συνάρτηση αξιολόγησης $(ED1+ED2)/2$:

Η χρησιμοποίηση του δείκτη $(ED1+ED2)/2$ ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή 0,62. Συγκεκριμένα τιμές $ED1=0,39$ και $ED2=0,85$ με τιμές παραμέτρων $P1=248$ και $P2=221$. Παρατηρείται ότι με την χρησιμοποίηση του δείκτη αυτού το άθροισμα των παραμέτρων είναι το ίδιο με το άθροισμα που προκύπτει από την χρησιμοποίηση του δείκτη $ED2$ ως συνάρτηση αξιολόγησης. Κατ' επέκταση το αποτέλεσμα της κατάτμησης είναι ακριβώς το ίδιο με την εικόνα 3.95, οπότε και δεν

εμφανίζεται ξανά. Παρατηρείται ότι ο ΓΑ, προσπαθώντας να μειώσει την τιμή του δείκτη $(ED1+ED2)/2$ στο ελάχιστο για να βελτιστοποιήσει τις παραμέτρους έδωσε ίδια τιμή στον ED2 με την βέλτιστη κατάτμηση με βάση αυτόν. Παρατηρείται δηλαδή ότι η χρήση του δείκτη αυτού δεν έδωσε διαφορετικό αποτέλεσμα και δεν έδρασε με διαφορετικό τρόπο, όπως είχε υλοποιηθεί με την εφαρμογή του αλγορίθμου κατάτμησης watershed.

3.5.4.3 Υλοποίηση στην ΔΕ της NASA για τον πλανήτη Άρη

Τα αποτελέσματα των κατατμήσεων που προέκυψαν χρησιμοποιώντας τον συγκεκριμένο αλγόριθμο κατάτμησης στην υλοποίηση του ΓΑ, εξαρτώνται από την συνάρτηση αξιολόγησης που χρησιμοποιείται κάθε φορά. Καλύτερα αποτελέσματα παρατηρούνται με την χρησιμοποίηση του δείκτη $(ED1+ED2)/2$, σε σχέση με τους υπόλοιπους δυο, ως προς το αποτέλεσμα της κατάτμησης, γεγονός όμοιο με την διαπίστωση για τον ανάλογο δείκτη στην ίδια εικόνα με εφαρμογή του αλγορίθμου κατάτμησης watershed. Ακολουθεί ο δείκτης ED2 και στο τέλος έρχεται ο δείκτης ED1.

Όσον αφορά τον χρόνο εκτέλεσης του ΓΑ αυτός κυμαίνεται από 8,7 ώρες έως 50 ώρες ανάλογα με τις επιλογές του ΓΑ και κυρίως με το πλήθος γενεών και τον εκάστοτε πληθυσμό. Για παράδειγμα με χρήση του δείκτη ED2 ως συνάρτηση αξιολόγησης καλύτερα αποτελέσματα προέκυψαν σε 19,5 ώρες στην δεύτερη περίπτωση του πίνακα 3.9. Ωστόσο υπήρξε και εφαρμογή του ΓΑ, με χρήση του δείκτη ED2, διάρκειας 32,9 ωρών που δεν απέδωσε βέλτιστη κατάτμηση.

Όσον αφορά τις επιλογές του ΓΑ στις διαδικασίες υλοποίησης του, επιλέχτηκαν σε κάθε εφαρμογή όσο το δυνατόν διαφορετικές, ώστε να ελεγχθούν τα αποτελέσματα που προκύπτουν. Με την χρήση του συγκεκριμένου αλγορίθμου, καθώς το εύρος τιμών των παραμέτρων ήταν το ίδιο, δόθηκε η δυνατότητα για χρήση περισσότερων επιλογών στην διαδικασία μετάλλαξης του ΓΑ. Ωστόσο στην διαδικασία της αρχικοποίησης εφαρμόστηκε η μέθοδος «Integer», ώστε να δημιουργούνται άτομα με ακέραιες τιμές.

Γενικότερα παρατηρείται ότι για την συγκεκριμένη εικόνα τα αποτελέσματα της κατάτμησης κρίνονται ικανοποιητικά για τους δυο από τους τρεις δείκτες (πλην ED1). Εντοπίζεται με ορθότητα και πληρότητα σε ποσοστό 90% το αντικείμενο αναφοράς. Επιπλέον παρατηρείται ότι ο χρόνος εκτέλεσης του ΓΑ, όπως και στον προηγούμενο αλγόριθμο, εξαρτάται από τις επιλογές στις διαδικασίες του ΓΑ. Ιδίως το πλήθος του πληθυσμού και των γενεών είναι οι καθοριστικοί παράγοντες του χρόνου εκτέλεσης του ΓΑ.

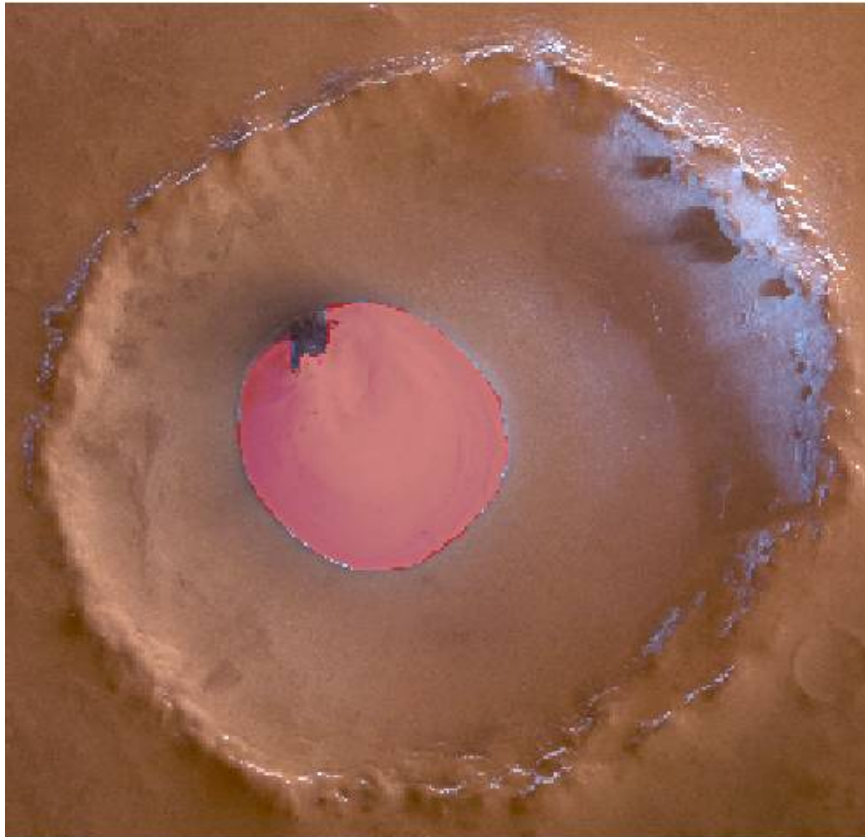
Για στατιστικούς λόγους αναφέρεται ότι ο συνολικός χρόνος εκτέλεσης των επτά εφαρμογών ανέρχεται σε περίπου 205 ώρες με μέσο όρο τις 29 ώρες.

ΔΕ NASA για τον πλανήτη Άρη 299pixel x 307pixel

Ημερ Έναρξης	Χρόνος Εκτ/σης	Pop Size	Initiali- zation	Selection	Crossover	Mutation	Gen	mutation rate	crossover rate	LB (P1_P2)	UB (P1_P2)	Συνάρτηση Αξιολόγησης	ΑΠΟΤΕΛΕΣΜΑΤΑ		
													P1	P2	Min Fitness /Min Raw
12/4/13	8,7h	40	Integer	Tourna- ment	Edge	SIM	30	2%	80%	1	500	ED2	47	6	1,21 / 1
14/4/13	19,5h	300	Integer	Uniform	Edge	SIM	10	2%	90%	1	500	ED2	17	4	2,70 / 0,02
17/4/13	50,8h	200	Integer	Uniform	Edge	Integer Gaussian	50	3%	80%	1	500	ED2	3	19	0 / 0,04
19/4/13	32,9h	500	Integer	Uniform	Edge	Swap	10	2%	85%	1	500	ED2	12	37	3,66 / 1,00
17/4/13	42h	100	Integer	Uniform	Cut Crossfill	SIM	80	2%	85%	1	450	ED1	167	146	0,005 / 0,0012
13/4/13	12h	200	Integer	Roulette Wheel	Uniform	Integer Gaussian	10	2%	90%	1	256	(ED1+ED2)/2	5	20	1,08 / 0,07
19/4/13	40h	100	Integer	Roulette Wheel	Two Point	Integer Range	80	2%	85%	1	450	(ED1+ED2)/2	8	29	0,88 / 0,07

Πίνακας 3.9 - Στοιχεία εφαρμογής ΓΑ στην ΔΕ της NASA για τον πλανήτη Άρη, με αλγόριθμο κατάτμησης region merging

Οι βέλτιστες κατατμήσεις που προκύπτουν για την εικόνα, για κάθε μια από τις τρεις συνάρτησης αξιολόγησης, με βάση τα αποτελέσματα του πίνακα 3.9 παρουσιάζονται στις εικόνες παρακάτω. Στην εικόνα 3.103 παρουσιάζονται το αντικείμενο αναφοράς σε επικάλυψη με την εικόνα από τον δορυφόρο της NASA, για την εποπτική παρουσίαση αυτού.

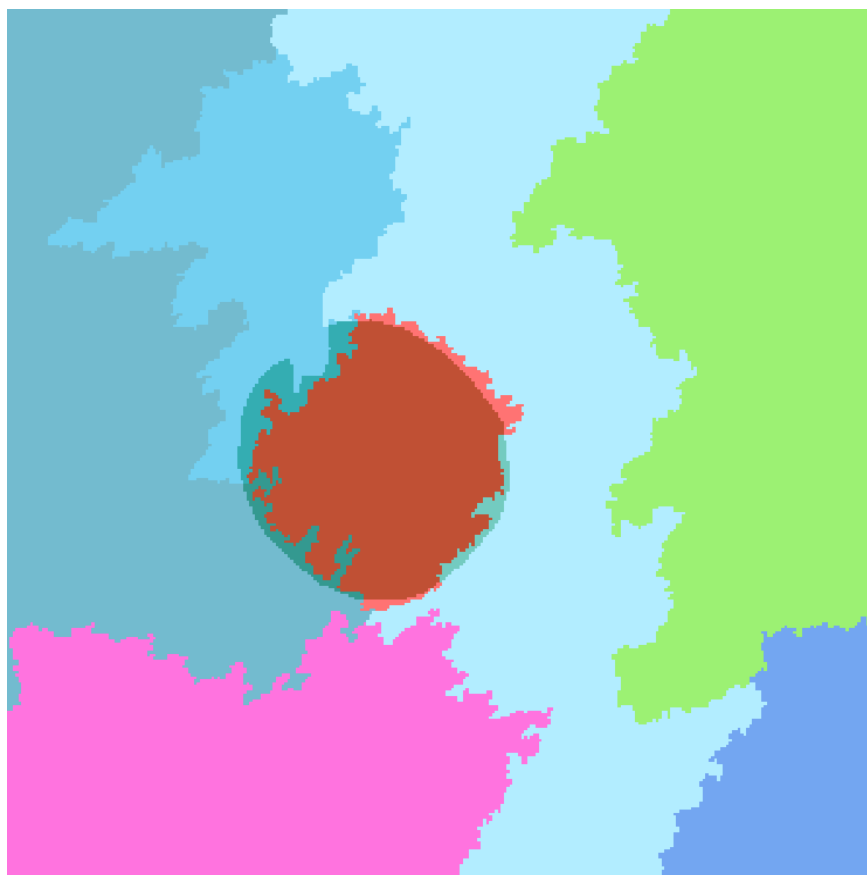


Εικόνα 3.103 - Αντικείμενο αναφοράς σε επικάλυψη με την εικόνα του πλανήτη Άρη

Συνάρτηση αξιολόγησης ED2:

Η χρησιμοποίηση του δείκτη ED2 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED2=0,02$ με τιμή καταλληλότητας 2,7. Ωστόσο υπήρξε και τιμή δείκτη 0,04 με τιμή καταλληλότητας 0. Στις εικόνες 3.104 και 3.105 παρουσιάζονται και οι δυο κατατμήσεις. Συγκεκριμένα για την πρώτη εφαρμογή οι τιμές των παραμέτρων ήταν $P1=17$ και $P2=4$, δηλαδή άθροισμα 21 ενώ με την δεύτερη εφαρμογή οι τιμές ήταν $P1=3$ και $P2=19$ δηλαδή άθροισμα 22. Ο δείκτης ED1 λαμβάνει τιμή 0.12 στην πρώτη περίπτωση και τιμή 0,115 στην δεύτερη περίπτωση. Παρατηρείται ότι καμιά από τις δυο τιμές αυτές δεν είναι η βέλτιστη με την χρησιμοποίηση του ED1 ως δείκτη για την συνάρτηση αξιολόγησης, καθώς η βέλτιστη τιμή είναι η $ED1=0,0012$. Το γεγονός αυτό αποδεικνύει για μια ακόμη φορά την διαφορετικότητα των δυο δεικτών μεταξύ τους.

Το πλήθος των αντικειμένων που δημιουργούνται και από τις δυο κατατμήσεις είναι επτά. Επιπλέον αν και ο δείκτης ED2 μηδενίστηκε στην μια κατάτμηση, το αντικείμενο αναφοράς δεν εντοπίστηκε αυτούσιο άλλα σε μεγάλο βαθμό, που σε ποσοστό ξεπερνάει το 90%. Ο δείκτης ED2 μπορεί να θεωρηθεί ότι λειτούργησε ορθά σε αντίθεση με την υλοποίηση στον αλγόριθμο watershed όπου δεν λειτούργησε σωστά. Ωστόσο παρατηρείται ότι δημιουργήθηκαν 7 αντικείμενα τα οποία ουσιαστικά αποτελούν σχεδόν ένα κοινό στην εικόνα 3.103, οπότε μια ορθή κατάτμηση θα έπρεπε να είχε περιορίσει περισσότερο το πλήθος αυτό.



Εικόνα 3.104 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (Region Merging - Min Fitness / Min Raw = 0 / 0,04)

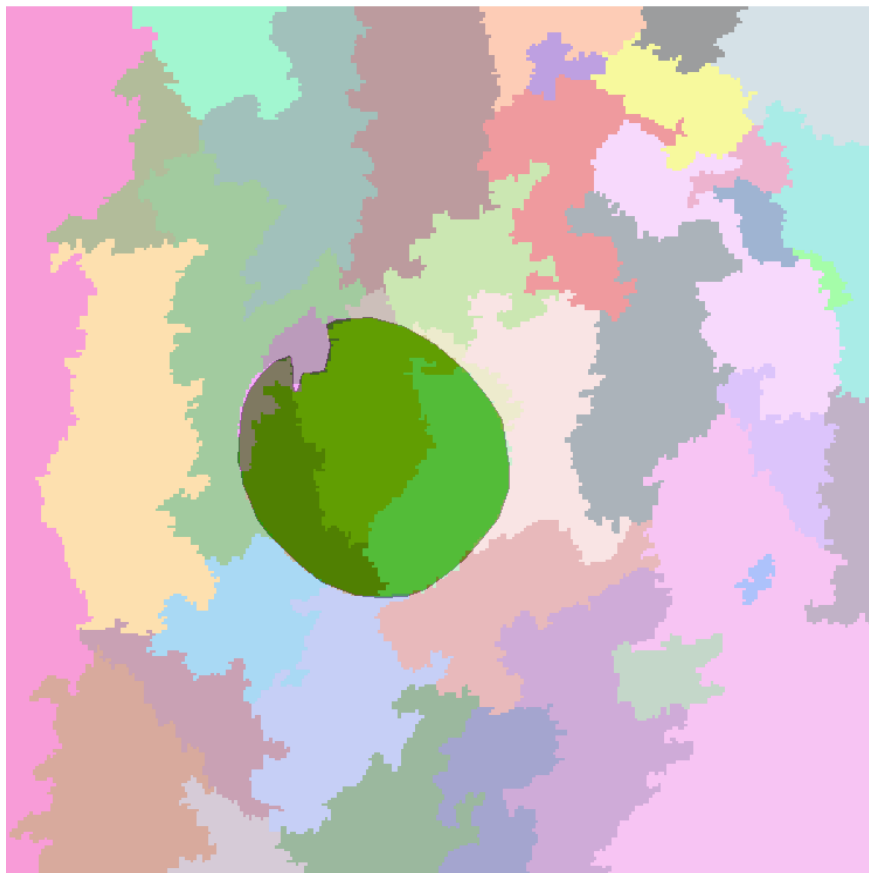


Εικόνα 3.105 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED2 (Region Merging - Min Fitness / Min Raw = 2,70 / 0,02)

Συνάρτηση αξιολόγησης ED1:

Η χρησιμοποίηση του δείκτη ED1 ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή $ED1=0,0012$ με τιμές παραμέτρων $P1=167$ και $P2=146$, δηλαδή άθροισμα 313. Παρατηρείται ότι ο δείκτης λειτούργησε διαφορετικά σε σχέση με τον τρόπο λειτουργίας του στις εικόνες του Αγ. Στεφάνου και στην περιοχή της Γερμανίας, όπου το αποτέλεσμα του ΓΑ ακολούθησε σχεδόν το αποτέλεσμα από την εφαρμογή του δείκτη ED2. Για τις τιμές αυτές ο δείκτης ED2 λαμβάνει τιμή $ED2=3$. Η ελάχιστη τιμή του δείκτη ED2, όπως ήδη αναφέρθηκε είναι 0,02, οπότε ενδυναμώνεται η διαφορετικότητα των δεικτών στην περίπτωση αυτή όπου υφίσταται ένα αντικείμενο αναφοράς.

Από την κατάτμηση δημιουργούνται συνολικά 43 αντικείμενα. Το μέγεθος αυτό είναι ιδιαίτερα μεγάλο και όπως παρουσιάζεται και στην εικόνα 3.106 η κατάτμηση δεν μπορεί να χαρακτηριστεί ως ορθή, λαμβάνοντας υπόψη το αντικείμενο αναφοράς, καθώς αυτό περιέχεται σε 4 αντικείμενα κατάτμησης. Ο ΓΑ ελαχιστοποιώντας τον δείκτη ED1, λειτούργησε μεν ορθά, όμως η ελαχιστοποίηση του δείκτη δεν αποδίδει τα βέλτιστα αποτελέσματα, καθώς δημιουργείται πλήθος αντικειμένων.



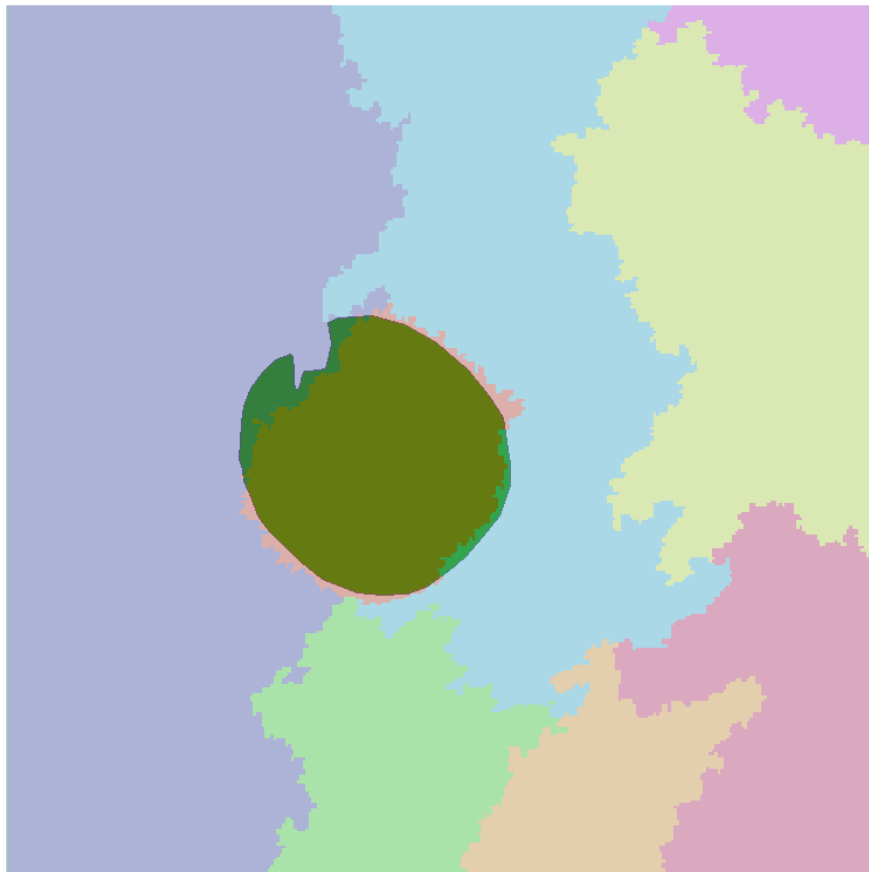
Εικόνα 3.106 - Αντικείμενα αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη ED1 (Region Merging - Min Fitness / Min Raw = 0,005 / 0,0012)

Συνάρτηση αξιολόγησης $(ED1+ED2)/2$:

Η χρησιμοποίηση του δείκτη $(ED1+ED)/2$ ως συνάρτηση αξιολόγησης έδωσε ως βέλτιστη τιμή 0,07. Συγκεκριμένα τιμές $ED1=0,088$ και $ED2=0,065$ με τιμές παραμέτρων $P1=8$ και $P2=29$ δηλαδή άθροισμα 37, διαφορετικό από τα αθροίσματα των εφαρμογών με τους δείκτες ED1 και ED2. Παρατηρείται ότι ο συνδυασμός των δυο δεικτών απέδωσε καλύτερα ως προς τον εντοπισμό του αντικειμένου αναφοράς, τόσο σε σχέση με την αποκλειστική χρήση του δείκτη ED1 όσο και του ED2. Το γεγονός αυτό έρχεται σε

αντίθεση με το συμπέρασμα που προέκυψε για τις δυο προηγούμενες εικόνες όπου παρατηρήθηκε ότι η χρήση του συνδυασμένου δείκτη απέφερε τα ίδια αποτελέσματα με τους δυο άλλους δείκτες και όχι καλύτερα.

Από την κατάτμηση δημιουργούνται συνολικά οκτώ αντικείμενα όσα και με τον δείκτη ED2. Παρατηρείται ότι το αντικείμενο αναφοράς έχει εντοπιστεί σε ίδιο σχεδόν ποσοστό με προηγουμένως όπου χρησιμοποιήθηκε ο δείκτης ED2, ενώ τα υπόλοιπα αντικείμενα της κατάτμησης διαφέρουν στα όρια τους, ιδίως στο αριστερό τμήμα της εικόνας. Η κατάτμηση μπορεί να χαρακτηριστεί ως βέλτιστη καθώς ανταποκρίνεται στην πραγματικότητα. Στην εικόνα 3.107 παρουσιάζεται το αποτέλεσμα της κατάτμησης σε επικάλυψη με το αντικείμενο αναφοράς.



Εικόνα 3.107 - Αντικείμενο αναφοράς σε επικάλυψη με το αποτέλεσμα της κατάτμησης, στην εικόνα του πλανήτη Άρη, με δείκτη $(ED1+ED2)/2$ (Region Merging - Min Fitness / Min Raw = 0,88 / 0,07)

4. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΡΟΤΑΣΕΙΣ

Στην παρούσα διπλωματική πραγματοποιήθηκε διερεύνηση της διαδικασίας κατάτμησης μιας εικόνας μέσω της εφαρμογής ΓΑ. Οι εικόνες που χρησιμοποιήθηκαν προέρχονταν από τον δορυφόρο IKONOS, από αερομεταφερόμενο σαρωτή Falcon II και από δορυφόρο της NASA. Για την υλοποίηση των ΓΑ χρησιμοποιήθηκαν τρεις διαφορετικοί αλγόριθμοι κατάτμησης, watershed, region growing και region merging μέσω των οποίων προέκυπτε το αποτέλεσμα της συνάρτησης αξιολόγησης για την οποία χρησιμοποιήθηκαν τρεις διαφορετικοί δείκτες ED1, ED2 και $(ED1+ED2)/2$. Σκοπός ήταν τόσο η εξέταση της ορθότητας των δεικτών ως προς την λειτουργικότητά τους, όσο και η βελτιστοποίηση των παραμέτρων που χρησιμοποιούν οι αλγόριθμοι κατάτμησης, ώστε να προκύψουν τα καλύτερα δυνατά αποτελέσματα. Κατόπιν όλων όσων αναφέρθηκαν στις παραγράφους 1 - 3, προκύπτουν τα εξής συμπεράσματα:

- Η εφαρμογή ΓΑ στον αλγόριθμο εντοπισμού κτιριακών εγκαταστάσεων **βελτιστοποίησε** τα αποτελέσματα των τιμών των παραμέτρων που χρησιμοποιούνται στον εκάστοτε αλγόριθμο. Σημαντική παράμετρος της διαδικασίας αυτής αποτέλεσε το γεγονός ότι δεν απαιτήθηκαν **καθόλου** δοκιμές από τον χρήστη ώστε να ελεγχθούν ποιοι συνδυασμοί παραμέτρων δίνουν έστω και κάποια ικανοποιητικά αποτελέσματα, διαδικασία που μπορεί να κρατήσει πολλές ώρες ή καλύτερα μέρες. Οι συνδυασμοί βέβαια αυτοί θα μπορούσαν να είναι άπειροι, αν αναλογιστεί κανείς ότι για παράδειγμα στον αλγόριθμο watershed, για τις τέσσερις παραμέτρους για τις οποίες εφαρμόστηκε ο ΓΑ και θεωρώντας ότι η πρώτη και η δεύτερη παράμετρος μπορεί να λάβει 250 διαφορετικές τιμές, η τρίτη 10 και η τέταρτη 4 (μπορεί να λάβουν και πολύ περισσότερες), τότε προκύπτουν $250*250*10*4=2.500.000$ διαφορετικοί συνδυασμοί. Βέβαια ο ΓΑ δεν έλεγξε όλους αυτούς τους συνδυασμούς για να καταλήξει στον βέλτιστο, αλλά ένα πολύ μικρό μέρος. Στο γεγονός αυτό βρίσκεται η **λειτουργικότητα** ενός ΓΑ, στο να καταλήξει στο βέλτιστο αποτέλεσμα χωρίς να εξετάσει όλους τους δυνατούς συνδυασμούς παραμέτρων. Τη σημαντικότερη θέση σε αυτή την διαδικασία ωστόσο, κατέχει όπως αναφέρθηκε σε προηγούμενα κεφάλαια, η **συνάρτηση αξιολόγησης**, η μη σωστή επιλογή της οποίας μπορεί να οδηγήσει σε ολέθρια αποτελέσματα. Επιπλέον πρέπει να δίνεται ιδιαίτερη προσοχή στο σκοπό της κάθε εργασίας ώστε να επιλέγεται και ανάλογη συνάρτηση αξιολόγησης.
- Σημαντική παράμετρος στην διαδικασία εφαρμογής του ΓΑ αποδείχτηκαν οι **επιλογές** που χρησιμοποιήθηκαν στις διάφορες δυνατότητες του. Πρέπει να δίνεται ιδιαίτερη **προσοχή** σε αυτές τις **επιλογές** δηλαδή στον τρόπο με τον οποίο θα γίνει η διασταύρωση, η μετάλλαξη, η επιλογή, ποιο θα είναι το μέγεθος του αρχικού πληθυσμού όπως επίσης και τα όρια στα οποία θα τερματιστεί ο ΓΑ αλλά και σε όλες τις υπόλοιπες δυνατότητες που αναλύθηκαν πρωτίτερα. Πριν επιλεγεί ποια τιμή θα χρησιμοποιηθεί σε κάθε δυνατότητα πρέπει να εξετάζεται το εκάστοτε πρόβλημα. Για παράδειγμα μέγεθος πληθυσμού μεγάλο, σε πρόβλημα με μικρό αριθμό παραμέτρων μπορεί να αποβεί **χρονοβόρο** χωρίς αποτελέσματα που να αξίζουν αυτή την σπατάλη χρόνου. Το ίδιο μπορεί να συμβεί εάν δοθούν πολύ μικρές τιμές στα όρια τερματισμού του ΓΑ σε ένα πρόβλημα που δεν ενδιαφέρει τόσο πολύ η ακρίβεια δεκαδικού. Για παράδειγμα στην εφαρμογή του ΓΑ με αλγόριθμο κατάτμησης region merging στην εικόνα της NASA, για τον δείκτη ED2, υπήρξε υλοποίηση με 10 γενιές και πληθυσμό 300 άτομα η οποία ολοκληρώθηκε σε 19,5 ώρες και υλοποίηση με πληθυσμό 200 άτομα αλλά με 50 γενιές η οποία ολοκληρώθηκε σε 50,8 ώρες.

Καλύτερα αποτελέσματα απέδωσε η πρώτη περίπτωση παρά τον λιγότερο χρόνο που χρειάστηκε ο ΓΑ για να ολοκληρωθεί (πίνακας 3.9). Το γεγονός αυτό οδηγεί στο συμπέρασμα ότι μεγαλύτερο πλήθος γενεών και πληθυσμού δεν συνεπάγεται απαραίτητα καλύτερη επίλυση του ΓΑ.

- Η επιλογή των αντικειμένων αναφοράς, που χρησιμοποιούνται στην διαδικασία υλοποίησης του ΓΑ, πρέπει να είναι ιδιαίτερως προσεκτική. Λανθασμένες επιλογές μπορεί να οδηγήσουν σε λάθος εφαρμογής του ΓΑ και κατ' επέκταση σε λανθασμένη βελτιστοποίηση των τιμών των παραμέτρων. Τα αντικείμενα αναφοράς πρέπει να ξεχωρίζουν από το γειτονικό περιβάλλον όσο το δυνατό περισσότερο και να ακολουθούν την ορθή γεωμετρία.
- Ο χρόνος υλοποίησης του ΓΑ εξαρτάται, εκτός από τις επιλογές του και από τις εξής παραμέτρους:
 - ✓ **Αλγόριθμος κατάτμησης,**
 - ✓ **Σύνολο** παραμέτρων του αλγορίθμου κατάτμησης,
 - ✓ **Εύρος τιμών** των παραμέτρων των αλγορίθμων κατάτμησης,
 - ✓ **Μέγεθος** της χρησιμοποιούμενης εικόνας,
 - ✓ **Δυσκολία** της χρησιμοποιούμενης εικόνας, καθότι εάν η διαδικασία της κατάτμησης είναι εύκολη σε μια εικόνα, ο ΓΑ υπάρχει περίπτωση να **τείνει** γρηγορότερα **στο αποτέλεσμα** και να τερματιστεί πλην την εξέταση όλων των επιλεγμένων γενιών,

δεδομένου βέβαια ότι οι επιλογές του ΓΑ είναι παρόμοιες, κυρίως όσον αφορά το πλήθος των γενεών και του πληθυσμού στις εφαρμογές του ΓΑ.
- Οι χρόνοι υλοποίησης των εφαρμογών του ΓΑ έχουν ως εξής:
 - Αλγόριθμος watershed:
 - ✓ Μικρότερος χρόνος υλοποίησης στην **ΔΕ της NASA** για τον **πλανήτη Άρη** με μέσο όρο ανά εφαρμογή στις **20,3 ώρες**, ακολουθεί η εικόνα **IKONOS του Αγ. Στεφάνου με 31 ώρες** και στο τέλος έρχεται η εικόνα από **αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας με χρόνο 44 ώρες**,
 - Αλγόριθμος region growing:
 - ✓ Μικρότερος χρόνος υλοποίησης στην **ΔΕ της NASA** για τον **πλανήτη Άρη** με μέσο όρο ανά εφαρμογή στις **70 ώρες**, ακολουθεί η εικόνα **IKONOS του Αγ. Στεφάνου με 83 ώρες** και στο τέλος έρχεται η εικόνα από **αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας με χρόνο 110 ώρες**,
 - Αλγόριθμος merging:
 - ✓ Μικρότερος χρόνος υλοποίησης στην **ΔΕ της NASA** για τον **πλανήτη Άρη** με μέσο όρο ανά εφαρμογή στις **29 ώρες**, ακολουθεί η εικόνα **IKONOS του Αγ. Στεφάνου με 57 ώρες** και στο τέλος έρχεται η εικόνα από **αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας με χρόνο 64 ώρες**.

Παρατηρείται ότι και για τις τρεις εικόνες η **ακολουθία χρόνου ήταν η ίδια** και στους τρεις αλγορίθμους. **Μικρότερο χρόνο** χρειαζόταν η **ΔΕ της NASA** για τον πλανήτη Άρη καθώς ήταν η μικρότερη σε διαστάσεις (299 x 307 pixel), στην συνέχεια **ακολουθεί η εικόνα IKONOS του Αγ. Στεφάνου** όπου οι διαστάσεις της ήταν οι μεγαλύτερες από τις τρεις εικόνες (501 x 501 pixel) και **τελευταία** έρχεται η εικόνα από **αερομεταφερόμενο σαρωτή για την περιοχή της Γερμανίας** (473 x 474 pixel). Προφανώς η **δυσκολία** της εικόνας από αερομεταφερόμενο σαρωτή ήταν μεγαλύτερη από την εικόνα IKONOS για αυτό χρειάστηκε και περισσότερο χρόνο και

για τους τρεις αλγόριθμους. Επιπλέον **γρηγορότερα** έδρασε ο αλγόριθμος κατάτμησης watershed, εν συνεχεία ο region merging και τελευταίος ο region growing, ανάλογα με την δυσκολία εφαρμογής τους.

- Ο δείκτης ελέγχου της κατάτμησης που πρότειναν οι Yong Liu et al (2012), ο οποίος λαμβάνει υπόψη του πλην των γεωμετρικών σχέσεων και τις αριθμητικές σχέσεις μεταξύ πολυγώνων αναφοράς και αντιστοιχισμένων πολυγώνων κατάτμησης, **δεν λειτουργεί ορθά** σε όλες τις περιπτώσεις εικόνων. Συγκεκριμένα στην εικόνα της NASA για τον πλανήτη Άρη, όπου είχε επιλεγεί ένα αντικείμενο αναφοράς, ο δείκτης ED2 δεν έδωσε ορθά αποτελέσματα. Μάλιστα υπήρξε περίπτωση όπου η **τιμή** του δείκτη ήταν **μηδέν**, δηλαδή **θεωρητικά** επιτεύχθηκε η **βέλτιστη κατάτμηση** μέσω αυτού, ωστόσο τα αποτελέσματα **δεν ήταν τα αναμενόμενα** (πίνακας 3.3 και εικόνα 3.48). Με την χρησιμοποίηση των αλγορίθμων κατάτμησης watershed και region merging, καλύτερα αποτελέσματα στην συγκεκριμένη εικόνα έδωσε ο **συνδυασμός** των δεικτών $(ED1+ED2)/2$. Για τον αλγόριθμο region growing καλύτερα αποτελέσματα έδωσε ο δείκτης ED1. Το γεγονός αυτό οδηγεί στο συμπέρασμα ότι όταν χρησιμοποιείται ένα αντικείμενο αναφοράς, **η αριθμητική σχέση** μεταξύ αυτού και των προκύπτων αντιστοιχισμένων πολυγώνων κατάτμησης **δεν είναι τόσο σημαντική** ώστε να επηρεάσει το αποτέλεσμα της κατάτμησης. Οι γεωμετρικές σχέσεις ωστόσο των πολυγώνων είναι σημαντικές και οδηγούν σε ορθή κατάτμηση. Παρατηρείται ότι ο δείκτης ED2 στις δυο από τις τρεις περιπτώσεις έδρασε ορθά με ταυτόχρονη χρήση του δείκτη ED1 και όχι μόνος του. Αντίθετα ο δείκτης ED1 έδρασε ορθά στην περίπτωση του αλγορίθμου region growing.
- Με χρήση του αλγορίθμου κατάτμησης **watershed** στις δυο εικόνες πλην της ΔΕ της NASA, **καλύτερα αποτελέσματα** κατάτμησης προέκυψαν με **την χρήση του δείκτη ED2** ως συνάρτηση αξιολόγησης. Με χρήση του αλγορίθμου κατάτμησης **region growing**, τα αποτελέσματα ήταν **όμοια** όπως πριν **για την εικόνα IKONOS του Αγ. Στεφάνου**. Αντίθετα στην εικόνα από **αερομεταφερόμενο σαρωτή** τα αποτελέσματα της κατάτμησης **ήταν τα ίδια και για τους τρεις δείκτες**. Μπορεί δηλαδή να πει κανείς ότι και πάλι ο αλγόριθμος ED2 λειτούργησε ορθά. Όσον αφορά όμως τον αλγόριθμο κατάτμησης **region merging** παρατηρείται ότι για τις δυο αυτές εικόνες τα αποτελέσματα της κατάτμησης **ήταν ακριβώς όμοια** και για τους χρησιμοποιηθέντες δείκτες. Μάλιστα στην εικόνα από αερομεταφερόμενο σαρωτή τα αποτελέσματα με χρήση του δείκτη ED2 ήταν ακριβώς ίδια με τα αποτελέσματα από την χρήση του δείκτη $(ED1+ED2)/2$. Τα γεγονότα αυτά οδηγούν στον συμπέρασμα ότι η χρήση του δείκτη ED2 δεν οδηγεί **καθολικά** σε διαφορετικά, καλύτερα αποτελέσματα κατάτμησης, από ότι προκύπτουν με χρήση του δείκτη ED1, αλλά ορισμένες μόνο φορές σε σχέση με την εικόνα που χρησιμοποιείται και σε σχέση με τον αλγόριθμο κατάτμησης.
- Ο εντοπισμός **όλων** των αντικειμένων αναφοράς σε μια εικόνα καθίσταται ιδιαίτερα **δύσκολος**. Ορθός εντοπισμός του ενός μπορεί να συνεπάγεται λανθασμένο εντοπισμό του άλλου αντικειμένου αναφοράς. Εξάιρεση αποτελεί η περίπτωση χρήσης ενός ή δυο αντικειμένων αναφοράς όπου επιδιώκεται η ορθή εξεύρεση του πλήθους των αντικειμένων. Γενικότερα αυτό οφείλεται στις διαφορετικές συνθήκες όπου επικρατούν σε κάθε αντικείμενο και στον τρόπο λειτουργίας που εφαρμόζει ο εκάστοτε αλγόριθμος κατάτμησης για να υλοποιήσει την κατάτμηση.

- Συμπερασματικά αναφέρεται ότι η εφαρμογή ΓΑ για την βελτιστοποίηση των παραμέτρων των αλγορίθμων κατάτμησης απέδωσε **ορθά αποτελέσματα** στις περισσότερες των περιπτώσεων. Επιπλέον ελέγχθηκε η ορθότητα του δείκτη ED2 και διαπιστώθηκαν τα προβλήματα που παρουσιάζει στην εκάστοτε χρήση του, ανάλογα με την εικόνα, τον αλγόριθμο κατάτμησης και το πλήθος των αντικειμένων αναφοράς.

Προτείνονται τα εξής:

- Εφαρμογή του ΓΑ σε **περισσότερες εικόνες**, τόσο παγχρωματικές (αλγόριθμος watershed) όσο και έγχρωμες, ώστε να ελεγχθούν και να επιβεβαιωθούν τα προηγούμενα συμπεράσματα σε μεγαλύτερο βαθμό. Επιπλέον οι εικόνες αυτές να προέρχονται και από άλλους δορυφόρους όπως QUICKBIRD, EROS, GEOEYE, LANDSAT. Θα μπορούσαν να χρησιμοποιηθούν ακόμα και αεροφωτογραφίες ώστε να ελεγχθεί η αποτελεσματικότητα του αλγορίθμου και σε αυτές.
- Έλεγχος του δείκτη ED2, με χρήση εικόνων στις οποίες να έχει επισημανθεί **ένα αντικείμενο αναφοράς** ώστε να ελεγχθεί το συμπέρασμα για μη ορθά αποτελέσματα κατάτμησης στην περίπτωση ελέγχου της κατάτμησης με χρήση του δείκτη αυτού. Επιπλέον θα μπορούσε να ελεγχθεί και για δυο αντικείμενα αναφοράς ώστε να αποδειχτεί **αν και μέχρι ποιο όριο πλήθους** αντικειμένων αναφοράς ο δείκτης ED2 δεν αποδίδει ορθά αποτελέσματα.
- Έλεγχος προκειμένου να διαπιστωθεί κατά πόσο μπορεί η παραπάνω μεθοδολογία να υλοποιηθεί στην **αντιμετώπιση** διαφόρων ειδών **προβλημάτων**. Για παράδειγμα να γίνεται σύγκριση μιας καμένης περιοχής σε δυο διαφορετικές χρονικές περιόδους ώστε να μπορεί να εντοπιστεί αυθαίρετη δόμηση και εν συνεχεία να πραγματοποιείται κατάτμηση των διαφορών των δυο εικόνων και εν συνεχεία ταξινόμηση των αποτελεσμάτων σε κατηγορίες ώστε να διαπιστώνεται τι αφορούν οι αλλαγές που έχουν επέλθει.
- Τέλος προτείνεται σε εικόνες από δορυφόρους με περισσότερα από τρία κανάλια, να χρησιμοποιηθεί **αντί** της αρχικής τριχρωμίας **RGB** στην διαδικασία της κατάτμησης, όπου ο αλγόριθμος το επιτρέπει κάποια **άλλη** τριχρωμία, ώστε να χρησιμοποιείται και το υπέρυθρο κανάλι. Σε περίπτωση χρήσης ενός καναλιού από τον αλγόριθμο κατάτμησης αυτό να είναι το υπέρυθρο. Επιπλέον θα μπορούσε να χρησιμοποιηθεί τριχρωμία η οποία να προέρχεται από μετατροπή της αρχικής τριχρωμίας σε κάποια διαφορετική. Για παράδειγμα θα μπορούσε να μετατραπεί σε HSL δηλαδή Hue, Saturation, Lightness, ή σε HSV δηλαδή Hue, Saturation, Value (Brightness) ή ακόμα και σε HIS δηλαδή Hue, Saturation, Intensity. Με τον τρόπο αυτό θα μπορούσε να υπάρξει μεγαλύτερος και λεπτομερέστερος έλεγχος της αποτελεσματικότητας των αλγορίθμων κατάτμησης, των δεικτών και κατ' επέκταση του ΓΑ.

ΒΙΒΛΙΟΓΡΑΦΙΚΕΣ ΠΗΓΕΣ

- [1] Αργιαλάς, Δ., 1998, “Ψηφιακή Τηλεπισκόπηση”, Εκδόσεις ΣΑΤΜ/ ΕΜΠ, Αθήνα.
- [2] Αργιαλάς, Δ., 1999, “Φωτοερμηνεία - Τηλεπισκόπηση”, Εκδόσεις ΣΑΤΜ/ΕΜΠ, Αθήνα.
- [3] Καραντζαλος Κ., 2007. “Αυτόματη εξαγωγή χαρακτηριστικών από εναέριες και δορυφορικές εικόνες με τεχνικές υπολογιστικής όρασης”, Διδακτορική Διατριβή.
- [4] Κουμιανάκης Γεώργιος, 2012. Διπλωματική εργασία για την ΣΑΤΜ στο ΕΜΠ.
- [5] Λυκοθανάσης Σπυριδων, 2001. Γενετικοί Αλγόριθμοι και Εφαρμογές, ΕΑΠ.
- [6] Argialas D. P. and Mavrantza O. D.. Comparison of Edge Detection and Hough Transform Techniques for the Extraction of Geologic Features. Laboratory of Remote Sensing, School of Rural and Surveying Engineering, National Technical University of Athens, Greece.
- [7] Argialas D. P. and Krishnamurthy S.. Detection of Lines and Circles in Maps and Engineering Drawings. Laboratory of Remote Sensing, School of Rural and Surveying Engineering, National Technical University of Athens, Greece.
- [8] Argialas D. P. and Charles Harlow A., 1990. Computational Image Interpretation Models: An Overview and a Perspective. Laboratory of Remote Sensing, School of Rural and Surveying Engineering, National Technical University of Athens, Greece.
- [9] Bagley J.D., 1967. The behavior of adaptive systems which employ genetic and correlation algorithms.
- [10] Belpaeme Tony, 1999. Evolution of visual feature detectors, in: Proceedings of 1st Conference on Evolutionary Computation in Image Analysis and Signal Processing, Göteborg, Sweden, pp. 1–10.
- [11] Benedek Csaba, Descombes Xavier and Zerubia Josiane, 2010. Building detection in a single remotely sensed image with a point process of rectangles. Ariana Project-Team INRIA/CNRS/UNSA, B.P. 93, 06902 Sophia Antipolis, France.
- [12] Bhanu B., Lin Y., 2002. Learning composite operators for object detection, in: Proceedings of the Conference on Genetic and Evolutionary Computation, NY, USA, pp. 1003–1010.
- [13] Bhanu Bir, Lin Yingqiang, 2004. Object detection in multi-modal images using genetic programming. College of Engineering, University of California, Riverside, CA 92521, USA.
- [14] Caselles V., Catta F., Dibos F., 1993. A geometric model for active contours in image processing, *Numerische Mathematik*, 66:1-31.
- [15] Caselles V., Kimmel R., and Sapiro G., 1997. Geodesic active contours, *International Journal. of Computer Vision*, 22(1):61-79.
- [16] Clinton, N., Holt, A., Scarborough, J., Yan, L., Gong, P., 2010. Accuracy Assessment measures for object-based image segmentation goodness. *Photogrammetric Engineering and Remote Sensing* 76 (3), 289–299.
- [17] De Jong K.A., 1975. Analysis of the behavior of a class of genetic adaptive systems
- [18] Goldberg D.E., 1989. Genetic Algorithm in Search, Optimization and Machine Learning, Addison Wesley Publishing company, Inc.
- [19] Hanbury's Allan PhD thesis CMM, 2002. Ecole des Mines de Paris, 2002
- [20] Harris C., Buxton B., 1996. Evolving edge detectors with genetic programming. Proceedings of the First Annual Conference on Genetic Programming, MIT Press, Cambridge, MA, USA, pp. 309–314.
- [21] Holland J.H., 1975. Adaptation in Natural and Artificial Systems, M.I.T. Press.

- [22] Howard D., Roberts S.C., Brankin R., 1999. Target detection in SAR imagery by genetic programming, in: *Advances in Engineering Software*, vol. 30, no. 5, Elsevierpp. 303–311.
- [23] Jeon Byoung-Ki, Jang Jeong-Hun, and Hong Ki-Sang, 2002. Road Detection in Spaceborne SAR Images Using a Genetic Algorithm. *IEEE Transactions on geosciences and Remote Sensing*, Vol. 40, No.1.
- [24] Kamal Hammouche, Moussa Diaf and Patrick Siarry, 2007. A multilevel automatic thresholding method based on a genetic algorithm for a fast image segmentation. *Computer Vision and Image Understanding* 109 (2008) 163–175.
- [25] Karantzalos K. and Paragios N., 2010. Large - Scale buildings reconstruction through information fusion and 3-D priors. *IEEE Transactions on geosciences and remote sensing*, vol. 48, no. 5.
- [26] Kass M., Witkin A., and Terzopoulos D., 1987. Snakes: Active contour models, *IEEE International Conference in Computer Vision*, pp.261-268.
- [27] Michalewicz Z., 1992. *Genetic Algorithms+Data Structures = Evolution programs*, Springer–Verlag, 2nd ed.
- [28] Mumford D. and Shah J., 1989. Optimal approximations by piecewise smooth functions and associated variational-problems. *Communications on Pure and Applied Mathematics*, 42(5), pp.577-685.
- [29] Osher S., Sethian J.A., 1998. Fronts Propagating with curvature dependent speed: Algorithms based on hamilton-jacobi formulations, *Journal of Computational Physics*, 79, pp.12-49.
- [30] Poli R., 1996. Genetic programming for feature detection and image segmentation, in: T.C. Fogarty (Ed.), *Evolutionary Computation*, Springer-Verlag, Berlin, Germany, pp. 110–125.
- [31] Richard Nock and Frank Nielsen, 2004. Statistical Region Merging. *Transactions on pattern analysis and machine intelligence*.
- [32] Santos J.A., Ferreira C.D., Torres R. da S., Gonçaves M.A., Lamparelli R.A.C., 2010. A relevance feedback method based on genetic programming for classification of remote sensing images. *Institute of Computing, University of Campinas, Campinas, SP, Brazil*.
- [33] Stanhope S.A., J.M. Daida, 1998. Genetic programming for automatic target classification and recognition in synthetic aperture radar imagery, in: *Proceeding of the Seventh Conference on Evolutionary Programming*, Springer-Verlag, Berlin, Germany, pp. 735–744.
- [34] Tackett Walter Alden, 1993. Genetic programming for feature discovery and image discrimination, in: *Proceedings of 5th International Conference on Genetic Algorithm*, Morgan Kaufmann, San Mateo, CA, USA, pp. 303–311.
- [35] Tseng Ming-Hseng, Chen Sheng-Jhe, Hwang Gwo-Haur, Shen Ming-Yu, 2007. A genetic algorithm rule-based approach for land-cover classification. *International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS)*.
- [36] Weidner, U., 2008. Contribution to the assessment of segmentation quality for remote sensing applications. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 37 (Part B7), pp. 479–484.
- [37] Wen-Bing Tao, Jin-Wen Tian, Jian Liu.2003. Image segmentation by three-level thresholding based on maximum fuzzy entropy and genetic algorithm. *State Key Laboratory for Image Processing and Intelligence Control, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China*
- [38] Yong Liu, Ling Bian, Yuhong Meng, Huanping Wang, Shifu Zhang, Yining Yang, Xiaomin Shao, 2012. Discrepancy measures for selecting optimal combination of parameter values in object-based image analysis. *ISPRS Journal of Photogrammetry and Remote Sensing* 68 (2012) 144–156

- [39] Zhang Mengjie and Ciesielski Victor, 1999. Genetic Programming for Multiple Class Object Detection. Department of Computer Science, Royal Melbourne Institute of Technology GPO Box 2476V, Melbourne Victoria 3001, Australia.

Url's

http://en.wikipedia.org/wiki/HSL_and_HSV

<http://www.genetic-programming.org/books.html#AGP>

<http://www.kerrywong.com/2009/05/07/canny-edge-detection-auto-thresholding>

http://pyevolve.sourceforge.net/0_6rc1

https://github.com/jlengrand/Tippy/blob/master/examples/region_growing.py

<http://www.lengrand.fr/2011/11/simple-region-growing-implementation-in-python>

<http://www.lix.polytechnique.fr/~schwander/python-srm>

http://en.wikipedia.org/wiki/Watershed_%28image_processing%29

ΠΑΡΑΡΤΗΜΑΤΑ

ΠΑΡΑΡΤΗΜΑ «Α»

Αποτελέσματα της εφαρμογής του ΓΑ με χρησιμοποίηση του αλγορίθμου κατάτμησης που βασίζεται στην μέθοδο του υδροκρίτη (Watershed)

➤ ΔΕ ΙΚΟΝΟΣ Αγ. Στεφάνου Αττικής

✓ ED2 - 10/02/2013

Total time elapsed: 186733.183 seconds.

- GenomeBase

Score: 0.714317

Fitness: 9.449141

Params: {'allele': - GAlleles

Homogeneous: False

List size: 4

Alleles:

Allele for 0 position:

- GAlleleRange

Real: False

Ranges Count: 1

Range List:

Range from [0] to [255]

Allele for 1 position:

- GAlleleRange

Real: False

Ranges Count: 1

Range List:

Range from [0] to [255]

Allele for 2 position:

- GAlleleRange

Real: False

Ranges Count: 1

Range List:

Range from [0] to [10]

Allele for 3 position:

- GAlleleList

List size: 3

Allele Options: [3, 5, 7]

}

Slot [Evaluator] (Count: 1)

Name: test - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerAllele - Weight: 0.50

Doc: Allele initialization function of G1DList

To use this initializer, you must specify the *allele* genome parameter with the

:class:`GAllele.GAlleles` instance.

```

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
  with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point
  .. warning:: You can't use this crossover method for lists with just
  one element.

- G1DList
  List size:      4
  List:           [114, 247, 3, 3]

```

✓ ED2 - 11/02/2013

```

Total time elapsed: 66122.740 seconds.
- GenomeBase
  Score:          0.704961
  Fitness:        0.819696

  Params:         {'allele': - GAlleles
  Homogeneous:    False
  List size:      4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [0] to [255]

Allele for 1 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [0] to [255]

Allele for 2 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [0] to [10]

Allele for 3 position:
- GAlleleList
  List size:      3
  Allele Options: [3, 5, 7]}
Slot [Evaluator] (Count: 1)

```

```

Name: test - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point
  .. warning:: You can't use this crossover method for lists with just
one element.

- G1DList
  List size:      4
  List:           [195, 229, 3, 5]

```

✓ ED2 - 13/02/2013

```

Total time elapsed: 89516.765 seconds.
- GenomeBase
  Score:          0.710836
  Fitness:        2.192291

  Params:         {'allele': - GAlleles
  Homogeneous:    False
  List size:      4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1

```

```

    Range List:
    Range from [1] to [9]

Allele for 3 position:
- GAlleleList
  List size:      3
  Allele Options: [3, 5, 7]
}

Slot [Evaluator] (Count: 1)
  Name: test - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
  parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers

  Accepts the *rangemin* and *rangemax* genome parameters, both
  optional. Also
  accepts the parameter *gauss_mu* and the *gauss_sigma* which
  respectively
  represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size: 4
  List:      [220, 243, 3, 3]

```

✓ ED1 - 15/02/2013

```

Total time elapsed: 108633.415 seconds.
- GenomeBase
  Score:      0.110410
  Fitness:    0.279530

  Params:     {'allele': - GAlleles
  Homogeneous: False
  List size:  4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:      False
  Ranges Count: 1
  Range List:
  Range from [1] to [255]

```



```

Allele for 1 position:
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [1] to [9]

Allele for 3 position:
- GAlleleList
  List size:     3
  Allele Options: [3, 5, 7]
}

Slot [Evaluator] (Count: 1)
  Name: test - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
  parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers

  Accepts the *rangemin* and *rangemax* genome parameters, both
  optional. Also
  accepts the parameter *gauss_mu* and the *gauss_sigma* which
  respectively
  represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size: 4
  List:      [22, 179, 0, 5]

```

✓ ED1 - 17/02/2013

```

Total time elapsed: 118921.724 seconds.
- GenomeBase
  Score:          0.250328
  Fitness:        0.288905

  Params:         {'allele': - GAlleles

```

```

    Homogeneous:      False
    List size:        4
    Alleles:

Allele for 0 position:
- GAlleleRange
  Real:              False
  Ranges Count:     1
  Range List:
  Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:              False
  Ranges Count:     1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:              False
  Ranges Count:     1
  Range List:
  Range from [1] to [9]

Allele for 3 position:
- GAlleleList
  List size:         3
  Allele Options:   [3, 5, 7]
}

Slot [Evaluator] (Count: 1)
  Name: test - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:        4
  List:             [41, 239, 1, 3]

```

✓ ED1 - 18/02/2013

Total time elapsed: 87123.745 seconds.

- GenomeBase

Score: 0.255140
Fitness: 0.229835

Params: {'allele': - GAlleles
Homogeneous: False
List size: 4
Alleles:

Allele for 0 position:

- GAlleleRange

Real: False
Ranges Count: 1
Range List:
Range from [1] to [255]

Allele for 1 position:

- GAlleleRange

Real: False
Ranges Count: 1
Range List:
Range from [1] to [255]

Allele for 2 position:

- GAlleleRange

Real: False
Ranges Count: 1
Range List:
Range from [1] to [15]

Allele for 3 position:

- GAlleleList

List size: 4
Allele Options: [3, 5, 7, 9]

}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerAllele - Weight: 0.50
Doc: Allele initialization function of G1DList

To use this initializer, you must specify the *allele* genome parameter with the

```
:class:`GAllele.GAlleles` instance.
```

Slot [Mutator] (Count: 1)

Name: G1DListMutatorAllele - Weight: 0.50
Doc: The mutator of G1DList, Allele Mutator

To use this mutator, you must specify the *allele* genome parameter with the

```
:class:`GAllele.GAlleles` instance.
```

```

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      4
  List:           [104, 189, 1, 5]

```

✓ (ED1+ED2)/2 - 19/02/2013

```

Total time elapsed: 78500.559 seconds.
- GenomeBase
  Score:           0.613473
  Fitness:         1.497204

  Params:          {'allele': - GAlleles
  Homogeneous:     False
  List size:       4
  Alleles:

```

Allele for 0 position:

```

- GAlleleRange
  Real:            False
  Ranges Count:   1
  Range List:
  Range from [1] to [255]

```

Allele for 1 position:

```

- GAlleleRange
  Real:            False
  Ranges Count:   1
  Range List:
  Range from [1] to [255]

```

Allele for 2 position:

```

- GAlleleRange
  Real:            False
  Ranges Count:   1
  Range List:
  Range from [1] to [15]

```

Allele for 3 position:

```

- GAlleleList
  List size:       4
  Allele Options: [3, 5, 7, 9]
}

```

```

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50

```

```

Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

```

To use this initializer, you must specify the **allele** genome parameter with the
`:class:`GAllele.GAlleles`` instance.

```

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
  with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      4
  List:           [192, 40, 3, 3]

```

✓ (ED1+ED2)/2 - 22/02/2013

```

Total time elapsed: 158721.769 seconds.
- GenomeBase
  Score:          0.600305
  Fitness:        1.146088

  Params:         {'allele': - GAlleles
  Homogeneous:    False
  List size:      4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [15]

Allele for 3 position:
- GAlleleList
  List size:      4
  Allele Options: [3, 5, 7, 9]
}

```

```
Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      4
  List:          [8, 247, 3, 3]
```

➤ ΔΕ από αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας

✓ ED2 - 24/02/2013

```

Total time elapsed: 112441.850 seconds.
- GenomeBase
  Score:          0.539465
  Fitness:        3.407599

  Params:         {'allele': - GAlleles
  Homogeneous:    False
  List size:      4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [15]

Allele for 3 position:
- GAlleleList
  List size:      4
  Allele Options: [3, 5, 7, 9]
}

Slot [Evaluator] (Count: 1)
  Name: test - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSwap - Weight: 0.50
  Doc: The mutator of G1DList, Swap Mutator

.. note:: this mutator is :term:`Data Type Independent`

```

```

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverSinglePoint - Weight: 0.50
  Doc: The crossover of G1DList, Single Point

  .. warning:: You can't use this crossover method for lists with just
one element.

- G1DList
  List size:      4
  List:           [181, 226, 3, 13]

```

✓ ED2 - 25/02/2013

```

Total time elapsed: 152777.407 seconds.
- GenomeBase
  Score:          0.438309
  Fitness:        1.506638

  Params:         {'allele': - GAlleles
  Homogeneous:    False
  List size:      4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [15]

Allele for 3 position:
- GAlleleList
  List size:      4
  Allele Options: [3, 5, 7, 9]
}

Slot [Evaluator] (Count: 1)
  Name: test - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

```


To use this initializer, you must specify the *allele* genome parameter with the

```
:class:`GAllele.GAlleles` instance.
```

Slot [Mutator] (Count: 1)

Name: G1DListMutatorAllele - Weight: 0.50

Doc: The mutator of G1DList, Allele Mutator

To use this mutator, you must specify the *allele* genome parameter with the

```
:class:`GAllele.GAlleles` instance.
```

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverUniform - Weight: 0.50

Doc: The G1DList Uniform Crossover

- G1DList

List size: 4

List: [185, 199, 2, 9]

✓ ED2 - 26/02/2013

Total time elapsed: 127237.661 seconds.

- GenomeBase

Score: 0.778287

Fitness: 0.000000

Params: {'allele': - GAlleles

Homogeneous: False

List size: 4

Alleles:

Allele for 0 position:

- GAlleleRange

Real: False

Ranges Count: 1

Range List:

Range from [1] to [255]

Allele for 1 position:

- GAlleleRange

Real: False

Ranges Count: 1

Range List:

Range from [1] to [255]

Allele for 2 position:

- GAlleleRange

Real: False

Ranges Count: 1

Range List:

Range from [1] to [15]

Allele for 3 position:

- GAlleleList

```

List size:          3
Allele Options:    [3, 5, 7]}
Slot [Evaluator] (Count: 1)
  Name: test - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
  parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
  with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:          4
  List:              [40, 194, 2, 5]

```

✓ ED2 - 28/02/2013

```

Total time elapsed: 255538.120 seconds.
- GenomeBase
  Score:            0.775036
  Fitness:          23.072813

  Params:           {'allele': - GAlleles
  Homogeneous:      False
  List size:        4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:             False
  Ranges Count:    1
  Range List:
  Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:             False
  Ranges Count:    1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange

```

```

    Real:                False
    Ranges Count:       1
    Range List:
    Range from [1] to [15]

Allele for 3 position:
- GAlleleList
  List size:           4
  Allele Options:     [3, 5, 7, 9]
}

Slot [Evaluator] (Count: 1)
  Name: test - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:           4
  List:                [167, 178, 2, 3]

```

✓ ED1 - 28/02/2013

```

Total time elapsed: 170360.338 seconds.
- GenomeBase
  Score:                0.188503
  Fitness:              0.224181

  Params:               {'allele': - GAlleles
  Homogeneous:          False
  List size:            4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:                 False
  Ranges Count:        1
  Range List:
  Range from [1] to [255]

```

Allele for 1 position:

```
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [1] to [255]
```

Allele for 2 position:

```
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [1] to [15]
```

Allele for 3 position:

```
- GAlleleList
  List size:     3
  Allele Options: [3, 5, 7]
}
```

Slot [Evaluator] (Count: 1)

Name: test - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerAllele - Weight: 0.50

Doc: Allele initialization function of G1DList

To use this initializer, you must specify the *allele* genome parameter with the
:class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorAllele - Weight: 0.50

Doc: The mutator of G1DList, Allele Mutator

To use this mutator, you must specify the *allele* genome parameter with the

:class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverUniform - Weight: 0.50

Doc: The G1DList Uniform Crossover

```
- G1DList
```

List size: 4

List: [167, 250, 1, 5]

✓ (ED1+ED2)/2 - 24/02/2013

Total time elapsed: 89612.765 seconds.

```
- GenomeBase
```

Score: 0.617176

Fitness: 21.443543

Params: {'allele': - GAlleles

Homogeneous: False

List size: 4

```

Alleles:

Allele for 0 position:
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [1] to [15]

Allele for 3 position:
- GAlleleList
  List size:     4
  Allele Options: [3, 5, 7, 9]
}
Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:     4
  List:          [225, 100, 2, 9]

```

✓ (ED1+ED2)/2 - 26/02/2013

Total time elapsed: 178650.985 seconds.

- GenomeBase

Score: 0.447702
Fitness: 2.887513

Params: {'allele': - GAlleles
Homogeneous: False
List size: 4
Alleles:

Allele for 0 position:

- GAlleleRange

Real: False
Ranges Count: 1
Range List:
Range from [1] to [255]

Allele for 1 position:

- GAlleleRange

Real: False
Ranges Count: 1
Range List:
Range from [1] to [255]

Allele for 2 position:

- GAlleleRange

Real: False
Ranges Count: 1
Range List:
Range from [1] to [15]

Allele for 3 position:

- GAlleleList

List size: 4
Allele Options: [3, 5, 7, 9]}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerAllele - Weight: 0.50

Doc: Allele initialization function of G1DList

To use this initializer, you must specify the *allele* genome parameter with the

```
:class:`GAllele.GAlleles` instance.
```

Slot [Mutator] (Count: 1)

Name: G1DListMutatorAllele - Weight: 0.50

Doc: The mutator of G1DList, Allele Mutator

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverTwoPoint - Weight: 0.50

Doc: The G1DList crossover, Two Point

- G1DList

List size: 4
List: [39, 122, 2, 3]

➤ ΔΕ NASA για τον πλανήτη Άρη

✓ ED2 - 05/03/2013

Total time elapsed: 70127.371 seconds.

- GenomeBase

Score: 0.000000

Fitness: 24.329651

Params: {'allele': - GAlleles

Homogeneous: False

List size: 4

Alleles:

Allele for 0 position:

- GAlleleRange

Real: False

Ranges Count: 1

Range List:

Range from [1] to [255]

Allele for 1 position:

- GAlleleRange

Real: False

Ranges Count: 1

Range List:

Range from [1] to [255]

Allele for 2 position:

- GAlleleRange

Real: False

Ranges Count: 1

Range List:

Range from [1] to [15]

Allele for 3 position:

- GAlleleList

List size: 4

Allele Options: [3, 5, 7, 9]

}

Slot [Evaluator] (Count: 1)

Name: test - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerAllele - Weight: 0.50

Doc: Allele initialization function of G1DList

To use this initializer, you must specify the *allele* genome parameter with the

```
:class:`GAllele.GAlleles` instance.
```

Slot [Mutator] (Count: 1)

Name: G1DListMutatorAllele - Weight: 0.50

Doc: The mutator of G1DList, Allele Mutator

To use this mutator, you must specify the *allele* genome parameter with the

```

: class: `GAllele.GAlleles` instance.
Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      4
  List:           [8, 245, 14, 7]

```

✓ ED2 - 06/03/2013_A

```

Total time elapsed: 29563.681 seconds.
- GenomeBase
  Score:           0.000000
  Fitness:         1.541772

  Params:          {'allele': - GAlleles
  Homogeneous:     False
  List size:       4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:            False
  Ranges Count:   1
  Range List:
  Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:            False
  Ranges Count:   1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:            False
  Ranges Count:   1
  Range List:
  Range from [1] to [15]

Allele for 3 position:
- GAlleleList
  List size:       4
  Allele Options: [3, 5, 7, 9]
}

Slot [Evaluator] (Count: 1)
  Name: test - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
  parameter with the

```



```

: class: `GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
  with the
  : class: `GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      4
  List:           [187, 243, 14, 7]

```

✓ ED2 - 06/03/2013_B

```

Total time elapsed: 152777.407 seconds.
- GenomeBase
  Score:           0.438309
  Fitness:         1.506638

  Params:         {'allele': - GAlleles
  Homogeneous:    False
  List size:      4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [15]

Allele for 3 position:
- GAlleleList
  List size:      4
  Allele Options: [3, 5, 7, 9]
}

```

```

Slot [Evaluator] (Count: 1)
  Name: test - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      4
  List:           [185, 199, 2, 9]

```

✓ ED2 - 07/03/2013

```

Total time elapsed: 46835.939 seconds.
- GenomeBase
  Score:          0.000000
  Fitness:        0.979740

  Params:         {'allele': - GAlleles
  Homogeneous:    False
  List size:      4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:           False

```

```

    Ranges Count:      1
    Range List:
    Range from [1] to [15]

Allele for 3 position:
- GAlleleList
  List size:          4
  Allele Options:    [3, 5, 7, 9]
}

Slot [Evaluator] (Count: 1)
  Name: test - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point

  .. warning:: You can't use this crossover method for lists with just
one element.

- G1DList
  List size:          4
  List:               [199, 242, 8, 5]

```

✓ ED1 - 19/05/2013

```

Total time elapsed: 41998.260 seconds.
- GenomeBase
  Score:              0.020865
  Fitness:            0.110672

  Params:             {'allele': - GAlleles
  Homogeneous:        False
  List size:          4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:               False
  Ranges Count:      1
  Range List:

```

```

    Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [1] to [15]

Allele for 3 position:
- GAlleleList
  List size:     3
  Allele Options: [3, 5, 7]
}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point

  .. warning:: You can't use this crossover method for lists with just
one element.

- G1DList
  List size:     4
  List:          [2, 138, 2, 5]

```

✓ ED1 - 20/05/2013

Total time elapsed: 37049.107 seconds.

- GenomeBase

Score: 0.016231
Fitness: 0.219809

Params: {'allele': - GAlleles
Homogeneous: False
List size: 4
Alleles:

Allele for 0 position:

- GAlleleRange

Real: False
Ranges Count: 1
Range List:
Range from [1] to [255]

Allele for 1 position:

- GAlleleRange

Real: False
Ranges Count: 1
Range List:
Range from [1] to [255]

Allele for 2 position:

- GAlleleRange

Real: False
Ranges Count: 1
Range List:
Range from [1] to [15]

Allele for 3 position:

- GAlleleList

List size: 3
Allele Options: [3, 5, 7]

}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerAllele - Weight: 0.50
Doc: Allele initialization function of G1DList

To use this initializer, you must specify the **allele** genome parameter with the

:class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorAllele - Weight: 0.50
Doc: The mutator of G1DList, Allele Mutator

To use this mutator, you must specify the **allele** genome parameter with the

:class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)

```
Name: G1DListCrossoverUniform - Weight: 0.50
Doc: The G1DList Uniform Crossover
```

```
- G1DList
  List size:      4
  List:           [29, 140, 1, 3]
```

✓ (ED1+ED2)/2 - 06/03/2013

Total time elapsed: 78944.780 seconds.

```
- GenomeBase
  Score:          0.019314
  Fitness:        0.382679

  Params:         {'allele': - GAlleles
  Homogeneous:    False
  List size:      4
  Alleles:
```

Allele for 0 position:

```
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]
```

Allele for 1 position:

```
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]
```

Allele for 2 position:

```
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [15]
```

Allele for 3 position:

```
- GAlleleList
  List size:      4
  Allele Options: [3, 5, 7, 9]
}
```

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerAllele - Weight: 0.50

Doc: Allele initialization function of G1DList

To use this initializer, you must specify the **allele** genome parameter with the

```
:class:`GAllele.GAlleles` instance.
```

```

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
  with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point

  .. warning:: You can't use this crossover method for lists with just
  one element.

- G1DList
  List size:      4
  List:           [22, 148, 15, 3]

```

✓ (ED1+ED2)/2 - 07/03/2013

```

Total time elapsed: 101502.156 seconds.
- GenomeBase
  Score:          0.017171
  Fitness:        0.323750

  Params:         {'allele': - GAlleles
  Homogeneous:    False
  List size:      4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [255]

Allele for 2 position:
- GAlleleRange
  Real:           False
  Ranges Count:  1
  Range List:
  Range from [1] to [15]

Allele for 3 position:
- GAlleleList
  List size:      3
  Allele Options: [3, 5, 7]

```

```

}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      4
  List:           [39, 138, 6, 3]

```

✓ (ED1+ED2)/2 - 08/03/2013

```

Total time elapsed: 78186.909 seconds.
- GenomeBase
  Score:          0.017195
  Fitness:        11.976979

  Params:         {'allele': - GAlleles
  Homogeneous:    False
  List size:      4
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:           False
  Ranges Count:   1
  Range List:
                    Range from [1] to [255]

Allele for 1 position:
- GAlleleRange
  Real:           False
  Ranges Count:   1
  Range List:
                    Range from [1] to [255]

Allele for 2 position:
- GAlleleRange

```



```

    Real:                False
    Ranges Count:       1
    Range List:
    Range from [1] to [15]

Allele for 3 position:
- GAlleleList
  List size:           3
  Allele Options:     [3, 5, 7]
}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorAllele - Weight: 0.50
  Doc: The mutator of G1DList, Allele Mutator

  To use this mutator, you must specify the *allele* genome parameter
with the
  :class:`GAllele.GAlleles` instance.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point

  .. warning:: You can't use this crossover method for lists with just
one element.

- G1DList
  List size:           4
  List:                [141, 251, 8, 3]

```

ΠΑΡΑΡΤΗΜΑ «Β»

Αποτελέσματα της εφαρμογής του ΓΑ με χρησιμοποίηση του αλγορίθμου κατάτμησης που βασίζεται στην μέθοδο της τοπικής ανάλυσης επέκτασης περιοχών (Region Growing)

➤ ΔΕ ΙΚΟΝΟΣ Αγ. Στεφάνου Αττικής

✓ ED2 - 17/05/2013

Total time elapsed: 242159.109 seconds.

- GenomeBase

Score: 101.972166

Fitness: 126.132335

Params: {'rangemax': 210, 'rangemin': 30}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerInteger - Weight: 0.50

Doc: Integer initialization function of G1DList

This initializer accepts the **rangemin** and **rangemax** genome parameters.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorIntegerGaussian - Weight: 0.50

Doc: A gaussian mutator for G1DList of Integers

Accepts the **rangemin** and **rangemax** genome parameters, both optional. Also

accepts the parameter **gauss_mu** and the **gauss_sigma** which respectively

represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverTwoPoint - Weight: 0.50

Doc: The G1DList crossover, Two Point

.. warning:: You can't use this crossover method for lists with just one element.

- G1DList

List size: 3

List: [73, 33, 52]

✓ ED2 - 25/05/2013

Total time elapsed: 421397.810 seconds.

- GenomeBase

Score: 101.972166

Fitness: 129.557528

Params: {'rangemax': 230, 'rangemin': 30}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

```

Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerRange - Weight: 0.50
  Doc: Simple integer range mutator for G1DList

  Accepts the *rangemin* and *rangemax* genome parameters, both
  optional.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point

  .. warning:: You can't use this crossover method for lists with just
  one element.

- G1DList
  List size:          3
  List:               [194, 46, 67]

```

✓ ED1 - 25/03/2013

```

Total time elapsed: 304237.921 seconds.
- GenomeBase
  Score:              0.702586
  Fitness:            0.436653

  Params:             {'rangemax': 220, 'rangemin': 20}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers

  Accepts the *rangemin* and *rangemax* genome parameters, both
  optional. Also
  accepts the parameter *gauss_mu* and the *gauss_sigma* which
  respectively
  represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point

```

```
.. warning:: You can't use this crossover method for lists with just
one element.
```

```
- G1DList
  List size:      3
  List:          [47, 143, 205]
```

✓ (ED1+ED2)/2 - 20/03/2013

```
Total time elapsed: 224228.514 seconds.
```

```
- GenomeBase
  Score:          51.344354
  Fitness:       67.473959

  Params:        {'rangemax': 230, 'rangemin': 30}
```

```
Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
```

```
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList
```

This initializer accepts the **rangemin** and **rangemax** genome parameters.

```
Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSIM - Weight: 0.50
  Doc: The mutator of G1DList, Simple Inversion Mutation
```

```
.. note:: this mutator is :term:`Data Type Independent`
```

```
Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover
```

```
- G1DList
  List size:      3
  List:          [209, 57, 84]
```

➤ ΔΕ από αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας

✓ ED2 - 03/05/2013

```

Total time elapsed: 534508.111 seconds.
- GenomeBase
  Score:          42.865911
  Fitness:       131.987077

  Params:        {'allele': - GAlleles
  Homogeneous:   False
  List size:     3
  Alleles:

Allele for 0 position:
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [10] to [220]

Allele for 1 position:
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [30] to [200]

Allele for 2 position:
- GAlleleRange
  Real:          False
  Ranges Count: 1
  Range List:
  Range from [30] to [200]}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
  parameter with the
  :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSIM - Weight: 0.50
  Doc: The mutator of G1DList, Simple Inversion Mutation

  .. note:: this mutator is :term:`Data Type Independent`

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point

  .. warning:: You can't use this crossover method for lists with just
  one element.

```

```
- G1DList
  List size:      3
  List:          [67, 189, 50]
```

✓ ED2 - 22/05/2013

Total time elapsed: 425617.554 seconds.

```
- GenomeBase
  Score:          49.164539
  Fitness:        42.085608

  Params:         {'rangemax': 200, 'rangemin': 30}
```

```
Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
```

```
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList
```

This initializer accepts the **rangemin** and **rangemax** genome parameters.

```
Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers
```

Accepts the **rangemin** and **rangemax** genome parameters, both optional. Also accepts the parameter **gauss_mu** and the **gauss_sigma** which respectively represents the mean and the std. dev. of the random distribution.

```
Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverSinglePoint - Weight: 0.50
  Doc: The crossover of G1DList, Single Point
```

.. warning:: You can't use this crossover method for lists with just one element.

```
- G1DList
  List size:      3
  List:          [95, 42, 163]
```

✓ ED1 - 23/05/2013

Total time elapsed: 338402.844 seconds.

```
- GenomeBase
  Score:          0.700602
  Fitness:        0.629817

  Params:         {'rangemax': 200, 'rangemin': 50}
```

```
Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
```

```
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList
```

This initializer accepts the **rangemin** and **rangemax** genome parameters.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorSIM - Weight: 0.50

Doc: The mutator of G1DList, Simple Inversion Mutation

.. note:: this mutator is :term:`Data Type Independent`

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverTwoPoint - Weight: 0.50

Doc: The G1DList crossover, Two Point

.. warning:: You can't use this crossover method for lists with just one element.

- G1DList

List size: 3

List: [123, 149, 129]

✓ ED1 - 26/05/2013

Total time elapsed: 247813.815 seconds.

- GenomeBase

Score: 0.697

Fitness: 0.608

Params: {'rangemax': 200, 'rangemin': 30}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerInteger - Weight: 0.50

Doc: Integer initialization function of G1DList

This initializer accepts the **rangemin** and **rangemax** genome parameters.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorIntegerRange - Weight: 0.50

Doc: Simple integer range mutator for G1DList

Accepts the **rangemin** and **rangemax** genome parameters, both optional.

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverUniform - Weight: 0.50

Doc: The G1DList Uniform Crossover

- G1DList

List size: 3

List: [53, 31, 147]

✓ (ED1+ED2)/2 - 25/05/2013

Total time elapsed: 456354.579 seconds.

- GenomeBase

Score: 21.844921

Fitness: 22.375911

Params: {'rangemax': 200, 'rangemin': 20}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerInteger - Weight: 0.50

Doc: Integer initialization function of G1DList

This initializer accepts the **rangemin** and **rangemax** genome parameters.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorIntegerRange - Weight: 0.50

Doc: Simple integer range mutator for G1DList

Accepts the **rangemin** and **rangemax** genome parameters, both optional.

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverUniform - Weight: 0.50

Doc: The G1DList Uniform Crossover

- G1DList

List size: 3

List: [147, 172, 50]

➤ ΔΕ NASA για τον πλανήτη Άρη

✓ ED2 - 26/04/2013

Total time elapsed: 269414.922 seconds.

- GenomeBase

Score: 0.158066

Fitness: 1.278662

Params: {'rangemax': 230, 'rangemin': 20}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerInteger - Weight: 0.50

Doc: Integer initialization function of G1DList

This initializer accepts the *rangemin* and *rangemax* genome parameters.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorIntegerGaussian - Weight: 0.50

Doc: A gaussian mutator for G1DList of Integers

Accepts the *rangemin* and *rangemax* genome parameters, both optional. Also

accepts the parameter *gauss_mu* and the *gauss_sigma* which respectively

represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverEdge - Weight: 0.50

Doc: The Edge Recombination crossover for G1DList (widely used for TSP problem)

See more information in the `Edge Recombination Operator` http://en.wikipedia.org/wiki/Edge_recombination_operator Wikipedia entry.

- G1DList

List size: 3

List: [211, 158, 128]

✓ ED2 - 29/04/2013

Total time elapsed: 155431.703 seconds.

- GenomeBase

Score: 0.158363

Fitness: 4.548389

Params: {'allele': - GAlleles

Homogeneous: False

List size: 3

Alleles:

Allele for 0 position:

- GAlleleRange

```

    Real:                False
    Ranges Count:        1
    Range List:
    Range from [10] to [220]

Allele for 1 position:
- GAlleleRange
    Real:                False
    Ranges Count:        1
    Range List:
    Range from [30] to [200]

Allele for 2 position:
- GAlleleRange
    Real:                False
    Ranges Count:        1
    Range List:
    Range from [30] to [200]
}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerAllele - Weight: 0.50
  Doc: Allele initialization function of G1DList

  To use this initializer, you must specify the *allele* genome
  parameter with the
    :class:`GAllele.GAlleles` instance.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSIM - Weight: 0.50
  Doc: The mutator of G1DList, Simple Inversion Mutation

  .. note:: this mutator is :term:`Data Type Independent`

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverEdge - Weight: 0.50
  Doc: The Edge Recombination crossover for G1DList (widely used for
  TSP problem)

  See more information in the `Edge Recombination Operator
  <http://en.wikipedia.org/wiki/Edge\_recombination\_operator>`_
  Wikipedia entry.

- G1DList
  List size:            3
  List:                 [198, 173, 153]

```

✓ ED2 - 30/04/2013

Total time elapsed: 336009.003 seconds.

```

- GenomeBase
  Score:                0.158066
  Fitness:              3.442637

```

```

Params:          {'rangemax': 220, 'rangemin': 20}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers

  Accepts the *rangemin* and *rangemax* genome parameters, both
  optional. Also
  accepts the parameter *gauss_mu* and the *gauss_sigma* which
  respectively
  represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point

  .. warning:: You can't use this crossover method for lists with just
  one element.

- G1DList
  List size:      3
  List:           [99, 166, 138]

```

✓ ED1 - 10/05/2013

```

Total time elapsed: 217490.688 seconds.
- GenomeBase
  Score:          0.029074
  Fitness:        0.203146

  Params:          {'rangemax': 210, 'rangemin': 30}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers

  Accepts the *rangemin* and *rangemax* genome parameters, both
  optional. Also

```

```

accepts the parameter *gauss_mu* and the *gauss_sigma* which
respectively
represents the mean and the std. dev. of the random distribution.

```

```

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point

```

```

.. warning:: You can't use this crossover method for lists with just
one element.

```

```

- G1DList
  List size:      3
  List:           [71, 118, 112]

```

✓ (ED1+ED2)/2 - 17/05/2013

```

Total time elapsed: 282533.565 seconds.

```

```

- GenomeBase
  Score:          0.131151
  Fitness:        0.916774

  Params:         {'rangemax': 230, 'rangemin': 30}

```

```

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50

```

```

Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

```

```

This initializer accepts the *rangemin* and *rangemax* genome
parameters.

```

```

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSIM - Weight: 0.50
  Doc: The mutator of G1DList, Simple Inversion Mutation

```

```

.. note:: this mutator is :term:`Data Type Independent`

```

```

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverEdge - Weight: 0.50
  Doc: The Edge Recombination crossover for G1DList (widely used for
TSP problem)

```

```

See more information in the `Edge Recombination Operator
<http://en.wikipedia.org/wiki/Edge\_recombination\_operator>`_
Wikipedia entry.

```

```

- G1DList
  List size:      3
  List:           [174, 170, 145]

```

ΠΑΡΑΡΤΗΜΑ «Γ»

Αποτελέσματα της εφαρμογής του ΓΑ με χρησιμοποίηση του αλγορίθμου κατάτμησης που βασίζεται στην μέθοδο της τοπικής ένωσης περιοχών (Region Merging)

➤ ΔΕ ΙΚΟΝΟΣ Αγ. Στεφάνου Αττικής

✓ ED2 - 07/04/2013

Total time elapsed: 139997.346 seconds.

- GenomeBase

Score: 8.963664

Fitness: 17.192543

Params: {'rangemax': 250, 'rangemin': 1}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerInteger - Weight: 0.50

Doc: Integer initialization function of G1DList

This initializer accepts the *rangemin* and *rangemax* genome parameters.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorIntegerGaussian - Weight: 0.50

Doc: A gaussian mutator for G1DList of Integers

Accepts the *rangemin* and *rangemax* genome parameters, both optional. Also

accepts the parameter *gauss_mu* and the *gauss_sigma* which respectively

represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverUniform - Weight: 0.50

Doc: The G1DList Uniform Crossover

- G1DList

List size: 2

List: [165, 207]

✓ ED2 - 11/04/2013_A

Total time elapsed: 153131.460 seconds.

- GenomeBase

Score: 11.657311

Fitness: 14.500000

Params: {'rangemax': 256, 'rangemin': 1}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerInteger - Weight: 0.50

Doc: Integer initialization function of G1DList

```

This initializer accepts the *rangemin* and *rangemax* genome
parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSIM - Weight: 0.50
  Doc: The mutator of G1DList, Simple Inversion Mutation

  .. note:: this mutator is :term:`Data Type Independent`

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverEdge - Weight: 0.50
  Doc: The Edge Recombination crossover for G1DList (widely used for
TSP problem)

See more information in the `Edge Recombination Operator
<http://en.wikipedia.org/wiki/Edge\_recombination\_operator>`_
Wikipedia entry.

- G1DList
  List size:      2
  List:           [156, 220]

```

✓ ED2 - 11/04/2013_B

```

Total time elapsed: 210829.968 seconds.
- GenomeBase
  Score:           8.959748
  Fitness:         21.765318

  Params:          {'rangemax': 256, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

This initializer accepts the *rangemin* and *rangemax* genome
parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers

Accepts the *rangemin* and *rangemax* genome parameters, both
optional. Also
accepts the parameter *gauss_mu* and the *gauss_sigma* which
respectively
represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      2
  List:           [148, 239]

```

✓ ED1 - 12/04/2013

```

Total time elapsed: 167326.453 seconds.
- GenomeBase
  Score:          0.670086
  Fitness:        0.750000

  Params:         {'rangemax': 256, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSIM - Weight: 0.50
  Doc: The mutator of G1DList, Simple Inversion Mutation

  .. note:: this mutator is :term:`Data Type Independent`

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverCutCrossfill - Weight: 0.50
  Doc: The crossover of G1DList, Cut and crossfill, for permutations

- G1DList
  List size:      2
  List:           [162, 206]

```

✓ ED1 - 16/04/2013

```

Total time elapsed: 263165.334 seconds.
- GenomeBase
  Score:          0.669503
  Fitness:        0.289511

  Params:         {'rangemax': 250, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSIM - Weight: 0.50
  Doc: The mutator of G1DList, Simple Inversion Mutation

  .. note:: this mutator is :term:`Data Type Independent`

```

```

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverEdge - Weight: 0.50
  Doc:  The Edge Recombination crossover for G1DList (widely used for
TSP problem)

  See more information in the `Edge Recombination Operator
<http://en.wikipedia.org/wiki/Edge\_recombination\_operator>`_
  Wikipedia entry.

- G1DList
  List size:      2
  List:           [122, 244]

```

✓ (ED1+ED2)/2 - 11/04/2013

```

Total time elapsed: 189000.109 seconds.
- GenomeBase
  Score:          4.812766
  Fitness:        9.417087

  Params:         {'rangemax': 256, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc:  Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc:  A gaussian mutator for G1DList of Integers

  Accepts the *rangemin* and *rangemax* genome parameters, both
optional. Also
  accepts the parameter *gauss_mu* and the *gauss_sigma* which
respectively
  represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc:  The G1DList Uniform Crossover

- G1DList
  List size:      2
  List:           [240, 147]

```

✓ (ED1+ED2)/2 - 17/04/2013

```

Total time elapsed: 172975.281 seconds.
- GenomeBase
  Score:          4.087964
  Fitness:        7.738671

  Params:         {'rangemax': 500, 'rangemin': 1}

```



```

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSIM - Weight: 0.50
  Doc: The mutator of G1DList, Simple Inversion Mutation

  .. note:: this mutator is :term:`Data Type Independent`

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverEdge - Weight: 0.50
  Doc: The Edge Recombination crossover for G1DList (widely used for
  TSP problem)

  See more information in the `Edge Recombination Operator
  <http://en.wikipedia.org/wiki/Edge\_recombination\_operator>`_
  Wikipedia entry.

- G1DList
  List size:      2
  List:           [270, 428]

```

➤ ΔΕ από αερομεταφερόμενο σαρωτή Falcon II σε περιοχή της Γερμανίας

✓ ED2 - 07/04/2013_A

Total time elapsed: 395516.071 seconds.

- GenomeBase

Score: 0.971257

Fitness: 1.324978

Params: {'rangemax': 256, 'rangemin': 1}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerInteger - Weight: 0.50

Doc: Integer initialization function of G1DList

This initializer accepts the *rangemin* and *rangemax* genome parameters.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorIntegerRange - Weight: 0.50

Doc: Simple integer range mutator for G1DList

Accepts the *rangemin* and *rangemax* genome parameters, both optional.

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverTwoPoint - Weight: 0.50

Doc: The G1DList crossover, Two Point

.. warning:: You can't use this crossover method for lists with just one element.

- G1DList

List size: 2

List: [226, 223]

✓ ED2 - 07/04/2013_B

Total time elapsed: 171607.484 seconds.

- GenomeBase

Score: 0.854769

Fitness: 1.390540

Params: {'rangemax': 250, 'rangemin': 1}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerInteger - Weight: 0.50

Doc: Integer initialization function of G1DList

This initializer accepts the *rangemin* and *rangemax* genome parameters.

```

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers

  Accepts the *rangemin* and *rangemax* genome parameters, both
  optional. Also
  accepts the parameter *gauss_mu* and the *gauss_sigma* which
  respectively
  represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      2
  List:           [247, 227]

```

✓ ED2 - 17/04/2013

```

Total time elapsed: 281071.469 seconds.
- GenomeBase
  Score:          0.854115
  Fitness:        1.449513

  Params:         {'rangemax': 256, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerRange - Weight: 0.50
  Doc: Simple integer range mutator for G1DList

  Accepts the *rangemin* and *rangemax* genome parameters, both
  optional.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverTwoPoint - Weight: 0.50
  Doc: The G1DList crossover, Two Point

  .. warning:: You can't use this crossover method for lists with just
  one element.

- G1DList
  List size:      2
  List:           [238, 239]

```

✓ ED1 - 08/04/2013

Total time elapsed: 132129.437 seconds.

- GenomeBase

Score: 0.389533
Fitness: 0.409306

Params: {'rangemax': 256, 'rangemin': 1}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerInteger - Weight: 0.50

Doc: Integer initialization function of G1DList

This initializer accepts the **rangemin** and **rangemax** genome parameters.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorIntegerGaussian - Weight: 0.50

Doc: A gaussian mutator for G1DList of Integers

Accepts the **rangemin** and **rangemax** genome parameters, both optional. Also

accepts the parameter **gauss_mu** and the **gauss_sigma** which respectively

represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverUniform - Weight: 0.50

Doc: The G1DList Uniform Crossover

- G1DList

List size: 2

List: [241, 228]

✓ ED1 - 09/04/2013

Total time elapsed: 306768.139 seconds.

- GenomeBase

Score: 0.398292
Fitness: 0.354207

Params: {'rangemax': 256, 'rangemin': 1}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerInteger - Weight: 0.50

Doc: Integer initialization function of G1DList

This initializer accepts the **rangemin** and **rangemax** genome parameters.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorSIM - Weight: 0.50

Doc: The mutator of G1DList, Simple Inversion Mutation

```

.. note:: this mutator is :term:`Data Type Independent`

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverCutCrossfill - Weight: 0.50
  Doc: The crossover of G1DList, Cut and crossfill, for permutations

- G1DList
  List size:      2
  List:           [222, 190]

```

✓ ED1 - 10/04/2013

```

Total time elapsed: 107205.937 seconds.
- GenomeBase
  Score:           0.665784
  Fitness:         0.511798

  Params:          {'rangemax': 256, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers

  Accepts the *rangemin* and *rangemax* genome parameters, both
  optional. Also
  accepts the parameter *gauss_mu* and the *gauss_sigma* which
  respectively
  represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      2
  List:           [140, 247]

```

✓ (ED1+ED2)/2 - 09/04/2013

```

Total time elapsed: 233706.172 seconds.
- GenomeBase
  Score:           0.622282
  Fitness:         1.291295

  Params:          {'rangemax': 256, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50

```

```
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers

  Accepts the *rangemin* and *rangemax* genome parameters, both
  optional. Also
  accepts the parameter *gauss_mu* and the *gauss_sigma* which
  respectively
  represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      2
  List:          [248, 221]
```

➤ **ΔΕ NASA για τον πλανήτη Άρη**

✓ ED2 - 12/04/2013

```

Total time elapsed: 31320.163 seconds.
- GenomeBase
  Score:          1.025632
  Fitness:        1.20568

  Params:        {'rangemax': 500, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSIM - Weight: 0.50
  Doc: The mutator of G1DList, Simple Inversion Mutation

  .. note:: this mutator is :term:`Data Type Independent`

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverEdge - Weight: 0.50
  Doc: THE Edge Recombination crossover for G1DList (widely used for
  TSP problem)

  See more information in the `Edge Recombination Operator
  <http://en.wikipedia.org/wiki/Edge\_recombination\_operator>`_
  Wikipedia entry.

- G1DList
  List size:      2
  List:          [47, 6]

```

✓ ED2 - 14/04/2013

```

Total time elapsed: 70112.157 seconds.
- GenomeBase
  Score:          0.048932
  Fitness:        2.698127

  Params:        {'rangemax': 500, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

```

```

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSIM - Weight: 0.50
  Doc: The mutator of G1DList, Simple Inversion Mutation

  .. note:: this mutator is :term:`Data Type Independent`

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverEdge - Weight: 0.50
  Doc: THE Edge Recombination crossover for G1DList (widely used for
TSP problem)

  See more information in the `Edge Recombination Operator
<http://en.wikipedia.org/wiki/Edge\_recombination\_operator>`_
  Wikipedia entry.

- G1DList
  List size:      2
  List:           [17, 4]

```

✓ ED2 - 17/04/2013

```

Total time elapsed: 182809.297 seconds.
- GenomeBase
  Score:          0.047450
  Fitness:        0.000000

  Params:         {'rangemax': 450, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers

  Accepts the *rangemin* and *rangemax* genome parameters, both
optional. Also
  accepts the parameter *gauss_mu* and the *gauss_sigma* which
respectively
  represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverEdge - Weight: 0.50
  Doc: THE Edge Recombination crossover for G1DList (widely used for
TSP problem)

  See more information in the `Edge Recombination Operator
<http://en.wikipedia.org/wiki/Edge\_recombination\_operator>`_
  Wikipedia entry.

```



```
- G1DList
  List size:      2
  List:          [3, 19]
```

✓ ED2 - 19/04/2013

```
Total time elapsed: 118473.484 seconds.
- GenomeBase
  Score:          1.002155
  Fitness:        3.668532

  Params:         {'rangemax': 500, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSwap - Weight: 0.50
  Doc: The mutator of G1DList, Swap Mutator

  .. note:: this mutator is :term:`Data Type Independent`

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverEdge - Weight: 0.50
  Doc: The Edge Recombination crossover for G1DList (widely used for
  TSP problem)

  See more information in the `Edge Recombination Operator
  <http://en.wikipedia.org/wiki/Edge\_recombination\_operator>`_
  Wikipedia entry.

- G1DList
  List size:      2
  List:          [12, 37]
```

✓ ED1 - 17/04/2013

```
Total time elapsed: 153222.328 seconds.
- GenomeBase
  Score:          0.012180
  Fitness:        0.005716

  Params:         {'rangemax': 450, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
```

```

parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorSIM - Weight: 0.50
  Doc: The mutator of G1DList, Simple Inversion Mutation

  .. note:: this mutator is :term:`Data Type Independent`

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverCutCrossfill - Weight: 0.50
  Doc: The crossover of G1DList, Cut and crossfill, for permutations

- G1DList
  List size:      2
  List:           [167, 146]

```

✓ (ED1+ED2)/2 - 13/04/2013

```

Total time elapsed: 44562.328 seconds.
- GenomeBase
  Score:          0.074850
  Fitness:        1.087568

  Params:         {'rangemax': 256, 'rangemin': 1}

Slot [Evaluator] (Count: 1)
  Name: evaluation - Weight: 0.50
Slot [Initializer] (Count: 1)
  Name: G1DListInitializerInteger - Weight: 0.50
  Doc: Integer initialization function of G1DList

  This initializer accepts the *rangemin* and *rangemax* genome
  parameters.

Slot [Mutator] (Count: 1)
  Name: G1DListMutatorIntegerGaussian - Weight: 0.50
  Doc: A gaussian mutator for G1DList of Integers

  Accepts the *rangemin* and *rangemax* genome parameters, both
  optional. Also
  accepts the parameter *gauss_mu* and the *gauss_sigma* which
  respectively
  represents the mean and the std. dev. of the random distribution.

Slot [Crossover] (Count: 1)
  Name: G1DListCrossoverUniform - Weight: 0.50
  Doc: The G1DList Uniform Crossover

- G1DList
  List size:      2
  List:           [5, 20]

```

✓ (ED1+ED2)/2 - 19/04/2013

Total time elapsed: 144359.844 seconds.

- GenomeBase

Score: 0.076436
 Fitness: 0.885648

Params: {'rangemax': 450, 'rangemin': 1}

Slot [Evaluator] (Count: 1)

Name: evaluation - Weight: 0.50

Slot [Initializer] (Count: 1)

Name: G1DListInitializerInteger - Weight: 0.50

Doc: Integer initialization function of G1DList

This initializer accepts the **rangemin** and **rangemax** genome parameters.

Slot [Mutator] (Count: 1)

Name: G1DListMutatorIntegerRange - Weight: 0.50

Doc: Simple integer range mutator for G1DList

Accepts the **rangemin** and **rangemax** genome parameters, both optional.

Slot [Crossover] (Count: 1)

Name: G1DListCrossoverTwoPoint - Weight: 0.50

Doc: The G1DList crossover, Two Point

.. warning:: You can't use this crossover method for lists with just one element.

- G1DList

List size: 2

List: [8, 29]