Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας Τεχνολογιας Πληροφορικης Και Υπολογιστων

Διατμηματικο Προγραμμα Μεταπτυχιακων Σπουδων
Επιστημη Δεδομενων Και Μηχανικη Μαθηση

Audio-visual Self-Supervised Representation Learning in-the-wild

Μεταπτυχιακη Διπλωματικη Εργασια

Κωνσταντίνος Δ. Βιλουράς

Επιβλέπων: Στέφανος Κόλλιας
Καθηγητής Ε.Μ.Π

Συνεπιβλέπουσα: Παρασκευή Τζούβελη
Μέλος Ε.ΔΙ.Π.

Αθήνα, Ιούνιος 2022

Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας Τεχνολογιας Πληροφορικης Και Υπολογιστων

Audio-visual Self-Supervised Representation Learning in-the-wild

Μεταπτυχιακη Διπλωματικη Εργασια

Κωνσταντίνος Δ. Βιλουράς

Επιβλέπων: Στέφανος Κόλλιας
Καθηγητής Ε.Μ.Π

Συνεπιβλέπουσα: Παρασκευή Τζούβελη
Μέλος Ε.ΔΙ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 28η Ιουνίου 2022.

.....	.....	.....
Στέφανος Κόλλιας Καθηγητής Ε.Μ.Π.	Ανδρέας-Γεώργιος Σταφυλοπάτης Καθηγητής Ε.Μ.Π.	Παρασκευή Τζούβελη Μέλος Ε.ΔΙ.Π.

Αθήνα, Ιούνιος 2022

.....

Κωνσταντίνος Δ. Βιλουράς
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Α.Π.Θ.

Περίληψη

Τα τελευταία χρόνια παρατηρείται μια ραγδαία ανάπτυξη του κλάδου της μηχανικής μάθησης, με τα μοντέλα που προκύπτουν μέσω τεχνικών επιβλεπόμενης μάθησης να εφαρμόζονται ήδη σε διάφορους επιστημονικούς τομείς. Ωστόσο, η απαίτηση ύπαρξης ενός επισημασμένου συνόλου δεδομένων μεγάλης κλίμακας για την εκπαίδευση των μοντέλων αποτελεί περιοριστικό παράγοντα, καθώς η διαδικασία της επισήμανσης είναι ιδιαίτερα χρονοβόρα και κοστοβόρα. Εν αντιθέσει, οι αυτο-επιβλεπόμενες μέθοδοι μάθησης εξάγουν σήματα επίβλεψης απευθείας από τα δεδομένα με σκοπό την κωδικοποίηση της πληροφορίας που είναι απαραίτητη για να εξηγήσει τη δομή και τις ιδιότητές τους. Στην παρούσα διπλωματική εξετάζεται η περίπτωση των δεδομένων βίντεο, τα οποία φέρουν πληροφορία μέσω ενός συνδυασμού τροπικοτήτων (εικόνα και ήχος). Συγκεκριμένα, χρησιμοποιούνται βίντεο από μέσα κοινωνικής δικτύωσης τα οποία εμπεριέχουν θόρυβο, ενώ επίσης σε ένα μεγάλο ποσοστό αυτών υπάρχει μικρή συσχέτιση μεταξύ της οπτικής και της ακουστικής πληροφορίας. Επιπλέον, για τους σκοπούς της παρούσας ανάλυσης, χρησιμοποιούμε δύο μεθόδους αυτο-επιβλεπόμενης μάθησης. Η πρώτη μέθοδος βασίζεται στην τεχνική της συγκριτικής (contrastive) μάθησης, η οποία οδηγεί σε υψηλής ποιότητας οπτικο-ακουστικές αναπαραστάσεις. Αντίθετα, η δεύτερη μέθοδος, η οποία προτάθηκε πρόσφατα στη βιβλιογραφία, ανήκει στην κατηγορία των μη-συγκριτικών (non-contrastive) τεχνικών μάθησης και δεν έχει εφαρμοστεί ξανά σε οπτικοακουστικά δεδομένα. Τα πειραματικά αποτελέσματα σε δύο καθιερωμένα σύνολα δεδομένων δείχνουν την υπεροχή των πολυτροπικών μεθόδων μάθησης έναντι των αντίστοιχων μονοτροπικών. Επιπρόσθετα, η μέθοδος της συγκριτικής μάθησης οδηγεί σε σαφώς καλύτερα αποτελέσματα, καθώς αντιμετωπίζει σε μεγάλο βαθμό τα προβλήματα που δημιουργούνται από τα θορυβώδη δεδομένα, καθώς και από την αναντιστοιχία που προκύπτει στο ρυθμό εκπαίδευσης μεταξύ οπτικών και ακουστικών εισόδων. Επίσης, εξετάστηκε η δυνατότητα γενίκευσης των μοντέλων σε άγνωστα δεδομένα. Το συγκεκριμένο πείραμα έδειξε ότι τα αυτο-επιβλεπόμενα μοντέλα λειτουργούν καλύτερα σε τέτοιου είδους περιπτώσεις, οδηγώντας στο συμπέρασμα ότι δεν έχουν μοντελοποιήσει επαρκώς τα δεδομένα που ανήκουν στο σύνολο προ-εκπαίδευσης. Τέλος, μέσω της εφαρμογής του καλύτερου μας μοντέλου στο πρόβλημα της ανάκτησης βίντεο, καταλήγουμε στο ότι οι οπτικο-ακουστικές αναπαραστάσεις που προέκυψαν δεν είναι ικανοποιητικά ευθυγραμμισμένες, δηλαδή δεν έχει κωδικοποιηθεί επαρκώς η αντιστοίχιση εννοιών μεταξύ τροπικοτήτων.

Λέξεις κλειδιά

Βαθιά μάθηση, αυτο-επιβλεπόμενη μάθηση, αναγνώριση δράσεων σε βίντεο, ανάκτηση βίντεο

Abstract

In recent years, the field of machine learning has made tremendous progress in developing systems that can learn from large amounts of high-quality annotated data. Despite their success, it is clear that their performance is upper-bounded, as the vast majority of data available on the Internet is unlabeled and noisy, while the time and cost needed for the annotation process is prohibitive. Therefore, it is important to develop methods that allow networks to learn representations with limited supervision. Self-supervised learning has overcome these limitations by extracting learning signals from data alone. More specifically, through simple tasks such as predicting unobserved or hidden parts of an input, networks encode information about the underlying structure of data. As a result, this process yields powerful features that can be used in a variety of downstream tasks. In this study we focus on video signals, a rich data source which provides information in the form of naturally synchronized modalities, i.e. video and audio. In fact, we consider the case of data acquired in-the-wild, e.g. from social media platforms, which pose additional challenges such as weak audio-visual correspondences that typically occur in real-world scenarios. Furthermore, we use two self-supervised learning methods that are compatible with audio-visual inputs. The first is an established contrastive learning technique that shows promising results in popular action recognition benchmarks, whereas the second is a recently proposed non-contrastive approach that has not yet been applied to audio-visual data. We also compare both methods with their uni-modal counterparts to demonstrate the effectiveness of cross-modal learning. Results on a popular evaluation suite show that the contrastive learning technique outperforms all other methods, as it produces features which are both sufficiently linearly separable and also transferable across datasets. Moreover, the degradation in performance for the rest of the methods can be attributed to two factors, namely the lack of a mechanism that mitigates uninformative inputs, and also the difference in learning dynamics between visual and audio modality. Additionally, we present a novel benchmark for measuring generalization performance. The outcome of this experiment indicates that, although self-supervised models perform well on unseen concepts, they also seem to underfit the pre-training dataset. Last, we evaluate our top-performing model on video retrieval. For this task, we provide evidence that the model exhibits poor localization which, in turn, negatively affects cross-modal retrieval.

Keywords

deep learning, self-supervised learning, action recognition, video retrieval

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τον κ. Στέφανο Κόλλια για την ευκαιρία που μου έδωσε να ασχοληθώ με ένα τόσο ενδιαφέρον θέμα, καθώς και την κα. Παρασκευή Τζούβελη για τις συμβουλές και τη συνεχή υποστήριξή της καθ΄όλη τη διάρκεια εκπόνησης της παρούσας διπλωματικής. Επίσης, ευχαριστώ θερμά τον κ. Δημήτρη Ντελλή για την προθυμία του να βοηθήσει σε οποιοδήποτε πρόβλημα προέκυπτε σχετικά με την χρήση του υπερυπολογιστικού συστήματος ARIS. Τέλος, ένα μεγάλο ευχαριστώ σε συγγενείς και φίλους για την αμέριστη συμπαράστασή τους όλο αυτό το διάστημα.

Contents

List of Figures

List of Tables

Κεφάλαιο 0

Εκτεταμένη περίληψη στα Ελληνικά

0.1 Περιγραφή του προβλήματος

Ο κλάδος της μηχανικής μάθησης έχει γνωρίσει τεράστια άνθηση τα τελευταία χρόνια, με τα μοντέλα επιβλεπόμενης μάθησης να εφαρμόζονται σε πληθώρα προβλημάτων τόσο σε ερευνητικό επίπεδο όσο και στη βιομηχανία. Η επιτυχία των μεθόδων επιβλεπόμενης μάθησης οφείλεται κυρίως σε δύο παράγοντες, τη δυνατότητα των νευρωνικών δικτύων να μοντελοποιούν αποτελεσματικά ακόμα και δεδομένα υψηλής διαστατικότητας, καθώς και στην ύπαρξη επισημασμένων συνόλων δεδομένων μεγάλης κλίμακας.

Ωστόσο, η διαδικασία επισήμανσης των δεδομένων αποτελεί μια ιδιαίτερα χρονοβόρα και κοστοβόρα διαδικασία. Για παράδειγμα, ο χρόνος που απαιτείται για να κατηγοριοποιηθούν όλα τα διαθέσιμα δεδομένα στο διαδίκτυο είναι απαγορευτικά υψηλός, σε σημείο που είναι πρακτικά αδύνατο να υλοποιηθεί. Επιπλέον, λόγω και του προβλήματος των διφορούμενων επισημειώσεων, είναι πολύ πιθανό η παραπάνω διαδικασία να οδηγήσει σε υποβέλτιστα αποτελέσματα. Επομένως, θα πρέπει να αναπτυχθούν νέες, εξίσου αποτελεσματικές μέθοδοι που υποστηρίζουν δεδομένα ανεξάρτητα με το εάν αυτά είναι επισημασμένα ή όχι.

Πρόσφατες ερευνητικές προσπάθειες οδήγησαν στη δημιουργία μιας καινούριας οικογένειας τεχνικών που ονομάζονται αυτο-επιβλεπόμενες. Η ονομασία αυτή οφείλεται στο γεγονός ότι δεν απαιτούνται επισημειώσεις, καθώς το σήμα επίβλεψης προκύπτει απευθείας από τα δεδομένα. Χαρακτηριστικά παραδείγματα προβλημάτων αυτο-επιβλεπόμενης μάθησης αποτελούν η πρόβλεψη μελλοντικών παρατηρήσεων δεδομένου του παρελθόντος ή ακόμα και η ανακατασκευή μερών της εισόδου τα οποία δεν είναι ορατά από το δίκτυο. Με αυτό το τρόπο, τα νευρωνικά δίκτυα ανακαλύπτουν τη δομή των δεδομένων καθώς και χρήσιμες ιδιότητες τους, ενώ τα χαρακτηριστικά που προκύπτουν μπορούν να χρησιμοποιηθούν απευθείας είτε σε διαφορετικά σύνολα δεδομένων είτε σε νέα προβλήματα (π.χ. ταξινόμηση, αναγνώριση ή κατάτμηση).

Με βάση τα παραπάνω, είναι προφανές ότι τα δεδομένα βίντεο αποτελούν ιδανική επιλογή για αυτο-επιβλεπόμενη μάθηση, καθώς περιέχουν πλούσια πληροφορία με τη μορφή δύο διαφορετικών τροπικοτήτων (εικόνα και ήχος). Επιπλέον, λόγω της έλλειψης μεγάλων επισημασμένων συνόλων δεδομένων βίντεο, η εν λόγω ερευνητική περιοχή έχει επωφεληθεί σε μεγάλο βαθμό από τις αυτο-επιβλεπόμενες μεθόδους, σε σημείο που ξεπερνούν ακόμα και την επίδοση των επιβλεπόμενων μοντέλων. Επίσης, η πολυτροπική μάθηση αναπαραστάσεων οδηγεί σε εξίσου ισχυρές μονοτροπικές αναπαραστάσεις, οι οποίες μπορούν να εφαρμοστούν είτε σε προβλήματα όρασης υπολογιστών είτε σε αντίστοιχα προβλήματα επεξεργασίας ήχου.

Το θέμα της παρούσας διπλωματικής εργασίας αφορά την αυτο-επιβλεπόμενη μάθηση οπτικοακουστικών αναπαραστάσεων. Συγκεκριμένα, τα δεδομένα που χρησιμοποιούνται για

την εκπαίδευση προέρχονται από μέσα κοινωνικής δικτύωσης, επομένως εμπεριέχουν υψηλά επίπεδα θορύβου, καθώς και βίντεο στα οποία η οπτική και η ακουστική πληροφορία έχουν μικρή ή καθόλου συσχέτιση. Με αυτό τον τρόπο δημιουργούμε ένα δύσκολο σενάριο εκπαίδευσης το οποίο, εάν και έχει ήδη δειχθεί ότι περιορίζει την επίδοση των μοντέλων επιβλεπόμενης μάθησης, αντιπροσωπεύει σε μεγάλο βαθμό τα δεδομένα που συναντώνται συνήθως σε πραγματικές συνθήκες.

Στο Κεφάλαιο 2 παρουσιάζουμε τις δημοφιλέστερες μεθόδους αυτο-επιβλεπόμενης μάθησης οι οποίες προτάθηκαν στα πλαίσια της μάθησης οπτικών αναπαραστάσεων, καθώς και εκείνες τις μεθόδους που εφαρμόστηκαν με επιτυχία σε οπτικοακουστικά δεδομένα. Στη συνέχεια, στο Κεφάλαιο 3 εξηγούνται λεπτομερώς οι δύο μέθοδοι που επιλέχθηκαν στα πλαίσια της συγκεκριμένης εργασίας. Τα πειραματικά αποτελέσματα που προέκυψαν για το πρόβλημα της αναγνώρισης δράσεων σε βίντεο παρουσιάζονται στο Κεφάλαιο 4, ενώ στο Παράρτημα A απεικονίζονται τα αποτελέσματα στο πρόβλημα της ανάκτησης βίντεο. Τέλος, στο Κεφάλαιο 5 αναφέρουμε τα ερευνητικά μας ευρήματα, σχολιάζουμε τους πιθανούς λόγους για τους οποίους αυτά προέκυψαν, ενώ παράλληλα προτείνουμε και νέες κατευθύνσεις για μελλοντική έρευνα.

0.2 Συνεισφορές της εργασίας

Ο στόχος της παρούσας διπλωματικής εργασίας είναι η διερεύνηση των αυτο-επιβλεπόμενων μεθόδων μάθησης οπτικοακουστικών αναπαραστάσεων, η βαθύτερη κατανόηση της διαδικασίας εκπαίδευσης τους, καθώς και η εύρεση πιθανών αδύναμων σημείων τα οποία θα πρέπει να διευθετηθούν σε μελλοντικές εργασίες.

Για αυτό το σκοπό, επιλέξαμε να χρησιμοποιήσουμε δύο μεθόδους οι οποίες είναι συμβατές με πολυτροπικά δεδομένα. Η πρώτη μέθοδος αφορά μια δημοφιλή τεχνική που βασίζεται στη συγκριτική (contrastive) μάθηση και η οποία επιτυγχάνει υψηλά αποτελέσματα σε γνωστά σύνολα δεδομένων που αφορούν την αναγνώριση δράσεων σε βίντεο. Αντίθετα, η δεύτερη μέθοδος, η οποία προτάθηκε πρόσφατα στη σχετική βιβλιογραφία, στηρίζεται στη μη-συγκριτική (non-contrastive) μάθηση, ενώ επίσης δεν έχει εφαρμοστεί ξανά σε οπτικοακουστικά δεδομένα. Επιπλέον, συγκρίνουμε απευθείας τις πολυτροπικές μεθόδους με τις αντίστοιχες μονοτροπικές, δείχνοντας με αυτό τον τρόπο το πόσο αποτελεσματική είναι η μάθηση αναπαραστάσεων με χρήση πολλαπλών τροπικοτήτων.

Ακολουθώντας το καθιερωμένο πρωτόκολλο για την αξιολόγηση των αυτο-επιβλεπόμενων μοντέλων, παραθέτουμε τα αποτελέσματα για δύο είδη πειραμάτων που εκτελέστηκαν στα σύνολα UCF-101 και HMDB-51. Συγκεκριμένα, το πρώτο πείραμα εξετάζει το κατά πόσο τα χαρακτηριστικά που προέκυψαν είναι γραμμικά διαχωρίσιμα, ενώ στο δεύτερο πείραμα χρησιμοποιούμε τα βάρη των προ-εκπαιδευμένων δικτύων ως σημείο αρχικοποίησης και στη συνέχεια εκτελούμε μια διαδικασία μεταφοράς μάθησης (transfer learning). Επιπλέον, προτείνουμε ένα νέο τύπο πειράματος που εξετάζει τη δυνατότητα γενίκευσης των μοντέλων σε άγνωστες κλάσεις δεδομένων, δηλαδή σε κατηγορίες οι οποίες δεν υπάρχουν στο σύνολο δεδομένων προ-εκπαίδευσης.

Σύμφωνα με τα αποτελέσματα που προέκυψαν για τα δύο πρώτα πειράματα, παρατηρούμε ότι η τεχνική συγκριτικής μάθησης αποδίδει καλύτερα από όλες τις υπόλοιπες μεθόδους, συμπεριλαμβανομένου και ενός επιβλεπόμενου μοντέλου το οποίο συμπτύσσει την πληροφορία από τις οπτικές και τις ακουστικές εισόδους αντίστοιχα, κατά ένα μεγάλο ποσοστό. Ωστόσο, το γεγονός ότι οι υπόλοιπες μέθοδοι δεν λειτουργούν εξίσου καλά οφείλεται κυρίως σε δύο παράγοντες. Ο πρώτος παράγοντας αφορά την αντιμετώπιση των δεδομένων που εμπεριέχουν υψηλά επίπεδα θορύβου, ενώ ο δεύτερος παράγοντας σχετίζεται με το διαφορε-

τικό ρυθμό εκπαίδευσης ανά τύπο δεδομένων (για παράδειγμα, ένα δίκτυο που επεξεργάζεται ακουστικά δεδομένα εκπαιδεύεται πολύ πιο γρήγορα από ένα αντίστοιχο δίκτυο που εξάγει οπτικά χαρακτηριστικά).

Τα αποτελέσματα από το τρίτο πείραμα είναι εξίσου σημαντικά. Για παράδειγμα, δείξαμε ότι το πιο καθιερωμένο σύνολο δεδομένων για αυτο-επιβλεπόμενη μάθηση (Kinetics-400) έχει μεγάλη αλληλοεπικάλυψη με τα σύνολα αξιολόγησης UCF-101 και HMDB-51, άρα δεν παρουσιάζει σημαντική πληροφορία σχετικά με τη δυνατότητα γενίκευσης των μοντέλων. Επιπλέον, στη δική μας περίπτωση, καταλήγουμε στο συμπέρασμα ότι τα αυτο-επιβλεπόμενα μοντέλα επιτυγχάνουν υψηλές επιδόσεις σε δεδομένα που ανήκουν σε άγνωστες κατηγορίες, ωστόσο τα αποτελέσματα τους στις γνωστές κατηγορίες δεν είναι εξίσου ικανοποιητικά. Το συγκεκριμένο εύρημα καταδεικνύει την ανάγκη εξαγωγής χρήσιμων σημάτων εκπαίδευσης ακόμα και από θορυβώδη δεδομένα, καθώς η μη αντιμετώπιση τους ή η σταδιακή αφαίρεση τους κατά την εκπαίδευση θεωρούνται υποβέλτιστες λύσεις.

Τέλος, χρησιμοποιώντας το καλύτερο μοντέλο μας, εξετάστηκε η επίδοση του και στο πρόβλημα της ανάκτησης βίντεο. Στη συγκεκριμένη περίπτωση παρατηρούμε ότι ενώ τα αποτελέσματα της μονοτροπικής ανάκτησης (δηλαδή είτε με οπτική είτε με ακουστική πληροφορία) είναι ικανοποιητικά, δεν ισχύει το ίδιο και για την πολυτροπική ανάκτηση. Έτσι, καταλήγουμε στο συμπέρασμα ότι τόσο οι οπτικές όσο και οι ακουστικές αναπαραστάσεις δεν είναι επαρκώς ευθυγραμμισμένες, δηλαδή το μοντέλο δεν κατάφερε να κωδικοποιήσει την πληροφορία της αντιστοίχησης εννοιών μεταξύ τροπικοτήτων[1].

[1] Δηλαδή το γεγονός ότι ορισμένα είδη αντικειμένων παράγουν χαρακτηριστικούς ήχους, όπως συμβαίνει π.χ. στην περίπτωση των μουσικών οργάνων. Στο συγκεκριμένο παράδειγμα, το μοντέλο μπορεί να αντιληφθεί ότι ο ήχος που δέχεται στην είσοδο του προέρχεται από μουσικό όργανο, ωστόσο δεν είναι σε θέση να αντιστοιχήσει τον ήχο με την εικόνα του μουσικού οργάνου (χρώμα, σχήμα κλπ).

Chapter 1

Introduction

Machine learning has revolutionized research across a variety of disciplines. In addition, deep learning models have already been implemented and deployed to production for a wide spectrum of challenging tasks, ranging from healthcare [1] to monitoring nuclear reactor cores [2]. This indisputable success can be largely attributed to two key factors, namely the representation power of deep neural networks and the availability of large labeled datasets.

Data annotation, however, is an extremely tedious and time consuming process. More specifically, given the massive volume of data available on the Internet, it is practically infeasible to assign one or multiple labels to every single instance independently. Furthermore, label assignment is subject to annotator bias which, in turn, leads to ambiguous results. Therefore, it is essential to develop new, equally effective methods that overcome these limitations by being compatible with both unlabeled and labeled data.

Recently, self-supervised learning has emerged as a promising alternative to explicit supervision that does not require any labels. In fact, the fundamental idea is to extract learning signals from data itself in a meaningful way that would allow networks to discover their underlying structure and properties. Examples of self-supervised tasks involve predicting future observations given the past, reconstructing hidden parts of an input or even maximizing the similarity between a sample and a slightly perturbed view of itself. Those methods have been empirically shown to yield high-quality features that can be transferred across datasets and downstream tasks, while their performance is even comparable to supervised methods when pre-trained on large-scale data.

Video signals, in particular, are an ideal choice for self-supervised learning in a sense that they contain rich and diverse information in the form of co-occuring modalities, i.e. video and audio. Moreover, due to the absence of large annotated video datasets, this data source has benefited the most from self-supervision, showing results that are either competitive or even surpass supervised baselines. An additional advantage of cross-modal learning is that the network jointly learns powerful representations which can also be used for uni-modal applications, i.e. video- or audio-related tasks only.

Here, we consider the task of learning audio-visual representations via self-supervision. However, we primarily focus on the case of data acquired in-the-wild, for example from social media platforms, which typically consist of noisy (or missing) samples and uninformative video-audio pairs. This poses a challenging, yet realistic, learning scenario that negatively affects the learned representations' quality, thus highlighting the importance of enforcing robustness against erroneous inputs. Note that in-the-wild analysis is an ongoing challenge that has been explored in previous works, e.g. for human emotion

recognition [3]–[5].

The rest of this document is structured as follows. In Chapter 2 we detail the most popular self-supervised techniques for both visual and audio-visual representation learning, whereas in Chapter 3 we describe the two methods used in this study. Moreover, Chapter 4 presents our experimental results on an established action recognition benchmark, while additional results on the task of video retrieval can be found in Appendix A. Last, in Chapter 5 we discuss our findings and propose suggestions for future research.

1.1 Contributions

The goal of this study is to provide an in-depth analysis of audio-visual self-supervised methods, gain insight into their learning process, and also identify their weaknesses that should be addressed in future studies.

To this end, we use two methods that are compatible with multi-modal inputs. First, an established technique based on contrastive learning that achieves state-of-the-art results on the most popular action recognition benchmark. Second, a recently proposed method that has not been applied to audio-visual data before and which does not require any negative targets (unlike contrastive approaches). In addition, we juxtapose these two cross-modal methods with their uni-modal counterparts to demonstrate the effectiveness of learning with multiple modalities.

Following standard protocol, we evaluate the quality of the resulting representations on a popular benchmark suite consisting of two datasets, namely UCF-101 and HMDB-51. The first two experiments test the extent to which the learned features are linearly separable and transferable. Moreover, the third experiment provides a novel way to measure generalization performance, i.e. the models' ability to produce meaningful features even for inputs not previously encountered during self-supervised pre-training. To this end, given the labels in the pre-training dataset, we split categories in each downstream dataset into two disjoint sets, namely seen and unseen concepts, and then train a linear classifier for each set independently.

Results on the first two experiments show that the established cross-modal method outperforms all other approaches, including a supervised baseline that merges information from the visual and audio stream via late fusion, by a significant margin. This outcome, however, shed light into two important points, i.e. the necessity to mitigate noisy inputs to prevent performance degradation, and also the difference in learning dynamics between the visual and audio modality (the audio sub-network tends to overfit rather quickly). Note that the top-performing method explicitly avoids these issues by optimizing a loss that enforces robustness against uninformative data and by using memory targets during pre-training, respectively.

The findings from the third experiment provided more valuable insight. In fact, we have empirically shown that there is a high degree of class overlap (more than 80%) between a commonly used pre-training dataset (Kinetics-400) and this popular evaluation suite (UCF-101 and HMDB-51); therefore, it is not indicative of the models' true generalization performance. Furthermore, using our setup, we concluded that multi-modal self-supervised models perform well on unknown data domains, yet their accuracy on seen concepts is considerably lower. This result emphasizes the importance of extracting useful learning signals from noisy data as well, since neglecting them or simply discarding them is considered sub-optimal.

Last, although our top-performing model yields acceptable results on uni-modal video

retrieval, its performance on cross-modal retrieval is no better than random. This suggests that the corresponding video and audio features are not sufficiently aligned; therefore, the model exhibits poor localization.

Chapter 2

Background and related work

In this chapter we review the literature on self-supervised learning methods. In section 2.1, we introduce some of the most important techniques developed for the task of unsupervised visual representation learning[1] and attempt to group them into four distinct categories. In section 2.2, we describe those methods that were successfully extended to handle multi-modal inputs, i.e. both video and audio streams. Last, in section 2.3, we present the state-of-the-art on unsupervised video representation learning for both uni-modal and multi-modal approaches based on their performance on two established action recognition benchmarks.

2.1 Unsupervised visual representation learning

Common computer vision pipelines that employ self-supervised learning consist of two stages: the *pre-training* stage, where visual representations are learned without manually annotated data, and the *evaluation* stage, where the features extracted from the previous stage are used to solve real-world tasks such as recognition, detection and segmentation. In the following, we delve deeper into the pre-training stage and detail the methods that have been proposed for learning visual features in an unsupervised manner.

To compensate for the lack of labeled data, the key concept is to allow the network to discover the underlying structure of the data distribution, given that sample size is large enough. To this end, early studies focused on designing novel auxiliary tasks that would help the network to extract meaningful features and also encode useful invariances, e.g. with respect to rotation or color. However, since the choice of such tasks requires some level of domain expertise and also does not translate to other input modalities, recent works attempt to solve a much more straightforward task, i.e. maximizing the agreement between an input and an augmented view of itself based on a predefined similarity metric. As a result, we identify the following four types of learning visual features by using:

(i) auxiliary tasks, also known as pretext tasks

(ii) clustering-based methods

(iii) contrastive learning methods

(iv) non-contrastive learning methods

[1]Since self-supervised methods fall under the category of discriminative feature learning, we exclude all generative approaches from this overview.

2.1.1 Pretext tasks

As already mentioned, designing pretext tasks requires some level of supervision from the input data (except for labels), while the network needs to learn rich visual representations to successfully solve the task at hand. For example, Doersch et al. [6] extract random pairs of patches from each input image and task a neural network to predict the relative position of the second patch (query) given the first one (anchor). Here, spatial context serves as a strong supervisory signal that helps the network to identify high-level objects, as well as their constituent parts within an image. It is also worth mentioning that the authors of [6] identified *chromatic aberration*[2] as an effect that allows the network to solve the task by learning collapsed representations, e.g. a mapping to a constant vector. Therefore, to avoid such trivial shortcuts, it is suggested to use data augmentation schemes such as randomly dropping color channels or jittering each patch location by a certain amount of pixels. Along this line of work, Noroozi and Favaro [7] introduced the jigsaw puzzle task, i.e. the network learns to rearrange multiple patches based on their original location in the image. Moreover, Zhang et al. [8] formulate the task of predicting color information (ab channels) given the grayscale image (L channel) as a multinomial classification problem, while also enforcing class-rebalancing during training to encourage the network to learn rare colors as well. Pathak et al. [9] focus on inpainting, i.e. the network learns to fill in masked regions of the input given the surrounding pixels by optimizing a combination of a reconstruction and an adversarial loss. Zhang et al. [10] proposed split-brain autoencoders, two disjoint sub-networks that try to solve a cross-channel prediction task. Concretely, given a pair of diverse inputs such as (L, ab) channels or RGB channels and a depth map, each sub-network predicts the input of the other, which means that the concatenation of their outputs should be as close to the original image as possible. Last, Gidaris et al. [11] apply a random rotation to each input image (sampled from $\{0°, 90°, 180°, 270°\}$), which then serves as the model's target output during training.

2.1.2 Clustering-based methods

Clustering is a more general choice, since it requires little domain-specific knowledge, for grouping data that share the same visual attributes. In fact, clustering has already been studied in the context of supervised learning for high-level concept understanding [12]. Moreover, this type of contextual aggregation has additional advantages such as transparency and adaptation to novel data cases [13], which is crucial, e.g. for medical applications [14], [15].

Caron et al. [16] proposed DeepCluster, a two-stage pipeline that jointly learns the network's parameters and the cluster assignments of the resulting features. More specifically, the first step involves clustering all post-processed features (PCA reduced, whitened and L2-normalized) using k-means. Then, these assignments serve as *pseudo-labels*, which the network learns to predict in the second step by optimizing a standard cross-entropy loss. Moreover, the authors state that DeepCluster is prone to trivial solutions due to empty clusters that could lead to collapsed representations. To circumvent such issues, they follow two strategies: First, when empty clusters occur, they randomly sample a

[2] Chromatic aberration arises from the failure of the camera lens to focus colors onto the same focal point. This results in distorted color channels, i.e. an unwanted color might appear along the edges of objects. In the case of [6], a ConvNet learns the position of each patch relative to the lens by simply comparing the color channels, thus ignoring the rest of the information present in the image.

non-empty cluster and reassign its points into two new clusters using the original and a slightly perturbed version of its centroid. Second, they found that sampling images uniformly over the pseudo-labels is an effective way to avoid trivial parameterization. It is also worth mentioning that k-means computation time can be prohibitively long as it requires a forward pass on the entire dataset, which may consist of millions of samples. Asano et al. [17] proposed SelfLabel, a new framework that improved upon DeepCluster by enforcing an equi-partitioning constraint, as well as by adding heavier augmentation (e.g. random cropping and color jittering) during training to enforce invariance against non-semantic transformations. By introducing this type of constraint, they cast the label assignment problem to an instance of the optimal transport problem and can therefore be optimized using a fast version of the Sinkhorn-Knopp algorithm. In the framework of automatic data annotation, Ribeiro et al. [18] followed an alternative approach based on Bayesian neural networks. Concretely, they only add data with low label uncertainty to the final training set, while each sample has its own dedicated weight (inversely proportional to its predictive uncertainty) that influences gradient updates, i.e. learning process is mostly guided by examples with low uncertainty. In this case, clustering is used to distill and adapt knowledge between datasets without re-training networks.

2.1.3 Contrastive learning methods

Another category of methods that have shown promising results is based on contrastive learning. For example, Contrastive Predictive Coding [19] is a technique where the model learns to predict future samples given the latent representations of only past samples, up until the present step t. To do so, the authors formulate the noise-contrastive estimation loss (InfoNCE) which forces the latent space to capture only high-level information from the original signal, so that the model performs well on the future prediction task. They have also shown that minimizing InfoNCE loss maximizes a lower bound on mutual information, which means that the model progressively learns to preserve the mutual information between its latent representations and the original samples. Hénaff et al. [20] proposed an improved variant of CPC that uses larger models, newer normalization layers, heavier augmentation and bidirectional predictions, i.e. from past to future and vice versa. However, the most important component in both versions that is crucial for avoiding collapse is *negative sampling*, i.e. the model should both find the correct future sample and also learn to discriminate its representation for the current input from other, randomly drawn negative samples.

Wu et al. [21] introduced another important approach called Non-Parametric Instance Discrimination. Here, each sample of the dataset is considered as a separate "class" and the model performs a binary classification task, i.e. it learns to correctly identify the current sample from a set of negative samples that are randomly selected from the rest of the dataset. Also, as shown in [22], this method can be naturally extended to support multiple pairs of positives as well (e.g. complementary views of the same image or co-occuring modalities such as video frames and optical flow), which helps the model to learn more diverse and rich information. Moreover, adding more negatives to the learning objective is beneficial; however, it might be infeasible to do so likely due to large memory requirements. As a result, there appears to be a tradeoff between using "stale" features, i.e. from past iterations, and the number of negative samples that can be stored in memory. In [21] the authors use a memory bank to store the embeddings of the entire dataset (due to their small memory footprint) from the previous epoch. He et

al. [23] introduced MoCo as an alternative that is based on dictionary look-up. In fact, they keep an exponential moving average of the model weights as a momentum encoder, which is then used to encode both positive and negative targets used during training. As a result, this approach allows updating the dictionary through simple enqueuing/dequeuing operations that do not affect the mini-batch size. On the other end, SimCLR [24] performs negative mining on the current mini-batch directly, yet its performance is higher compared to the aforementioned techniques.

An additional novelty introduced in [24] is the use of a *projection head*, a two- or three-layer MLP that projects the model's representations to a lower-dimensional unit sphere (via L2 normalization), where subsequently the contrastive loss is applied. This module is only used during pre-training (the encoder's output is useful for downstream tasks) and has also yielded improved results when combined with MoCo [25]. In an attempt to demystify the contrastive objective, Wang et al. [26] show that, in the limit of infinite negative samples, optimizing the InfoNCE loss can be seen as maximizing both the similarity between positive samples (alignment term) and also the spread of the embeddings in the hypersphere (uniformity term) so that the resulting space is fully utilized.

There also exist hybrid methods such as SwAV [27], which combines clustering with contrastive learning. In particular, given a positive pair of images, the first step of SwAV involves projecting their features to the unit sphere. Then, an online clustering step is performed which maps the features to a set of trainable prototype vectors. This assignment results in two codes, one for each input, that are later fed to the swapped prediction mechanism, where the model learns to predict the code of the first view given the feature of the second view as input and vice versa. The loss function (cross-entropy) is defined in such a way that it can be jointly optimized both with respect to the prototypes, thus the online update of the clusters, and also the network parameters.

2.1.4 Non-contrastive learning methods

Although contrastive learning methods have achieved state-of-the-art results on various image recognition benchmarks, the underlying assumption that each instance of the dataset defines a unique class is too strong and does not necessarily hold in real world scenarios. This has motivated researchers to develop a new family of techniques that solely depend on positive pairs, which we refer to as non-contrastive methods.

For example, Grill et al. [28] introduced BYOL, an approach that directly maximizes the cosine similarity of positive pairs of samples. To this end, they define two networks, namely the online and the target network. The online network consists of an encoder, a projection head and a *prediction head* (the last two types of heads share the same architecture). On the other side, the target network can be seen as a copy of the online network that lacks a prediction head, while its weights are an exponential moving average of the online network's weights, thus they are not affected during backpropagation. The final learning objective involves the output of the predictor and the target network's projector, respectively. It is worth mentioning that this type of *asymmetry* in the overall model, as well as the *stop-gradient* operator (i.e. no parameter update is performed) that is applied to the target network, prevents the system from collapsing. Extending this line of work, Chen and He [29] showed that replacing the target network's weights with an actual copy of the online network's weights (excluding the predictor) at each iteration leads to faster convergence without sacrificing performance. Moreover, their experiments

confirm that both asymmetry and stop-gradient are the most crucial design choices that lead to stable training and also non-trivial solutions.

Furthermore, a recent work from Zbontar et al. [30] suggests that feature decorrelation is a technique that naturally avoids collapse without additional engineering tricks. In fact, they implemented a method, called Barlow Twins, that only uses an encoder and a projector module. Then, given a pair of positive samples, the model learns by pushing the cross-correlation matrix of the resulting representations towards the identity, thus effectively decorrelating each embedding dimension. They also found that decorrelation has a greater impact when performed on higher dimensions. Therefore, they modified the projector's architecture so that it maps its input to a higher dimensional space and also avoid the normalization step (mapping to the unit sphere). To distinguish it from the original projector, they name this module as the *expander*.

Building on the insights from all previous efforts, Bardes et al. [31] proposed an extension of Barlow Twins loss called VICReg. In particular, the new objective can be decomposed into the following three terms: variance regularization (for a given batch, maintain the variance of each embedding dimension above a certain threshold to avoid collapse), invariance regularization (given a positive pair of samples, minimize the distance between their embeddings) and covariance regularization (as in Barlow Twins, push off-diagonal elements of the cross-correlation matrix towards zero) term, respectively. In addition, this method also enables using asymmetric branches, i.e. each input can follow different paths, which is suitable for multi-modal inputs. In this case, VICReg loss can be applied to each branch independently and a direct comparison of the two outputs occurs only in the invariance regularization term.

For a direct comparison of some of the methods covered in this section, we present their schematics in Figure 2.1.

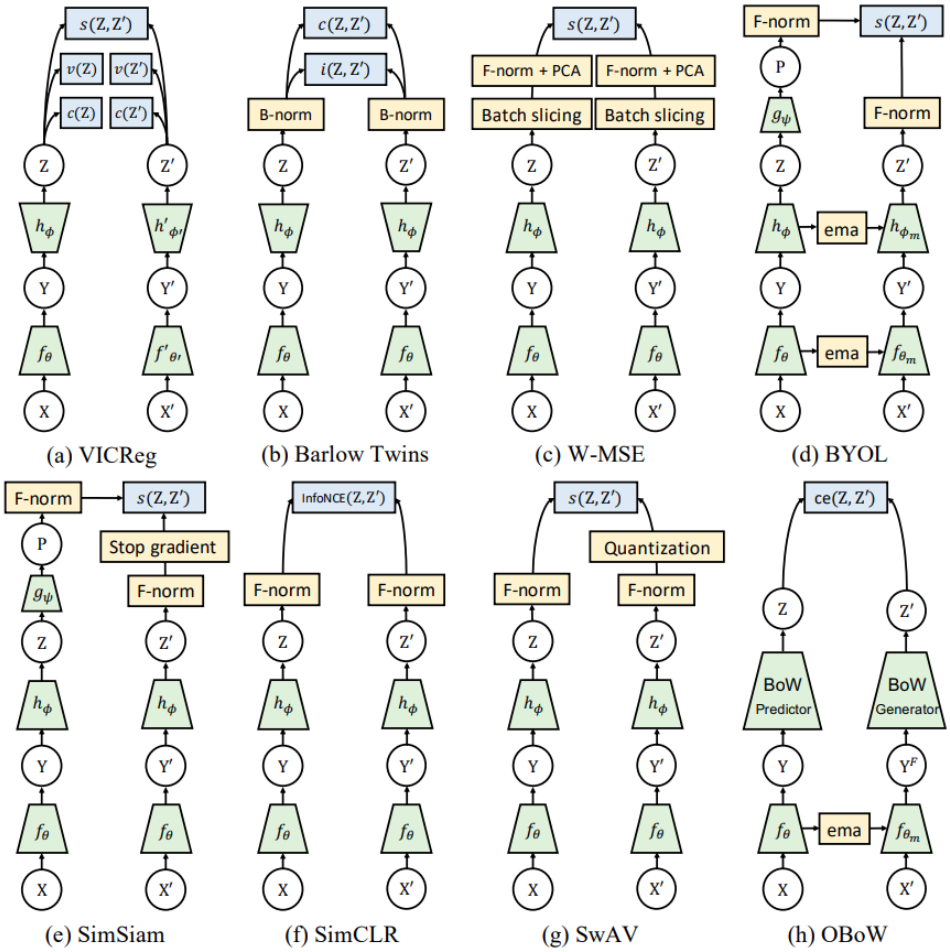

Figure 2.1: (taken from [31]) Overview of self-supervised methods found in the literature. Considering a pair of positive inputs X and X', an encoder f with weights θ produces the representations Y and Y', respectively. A projector module h maps the representations to another latent space, resulting in the embeddings Z and Z'. These embeddings are then used in the final learning criterion. For method (a) [31], the second branch can also support a different encoder f' and expander h' with weights θ' and ϕ', respectively. Each expander h and h' is depicted as a widening trapeze to denote a mapping to a higher dimensional space. The overall model is tasked to minimize a combination of regularizers v, c and s denoting variance, covariance and MSE loss, respectively. Method (b) [30] uses a decorrelation loss c and learns invariance through a loss i that makes similar dimensions highly correlated. Method (d) [28] has an asymmetric architecture, where the second branch uses an exponential moving average (ema) of the weights from the first branch and also lacks a predictor g which is only present in the first branch with weights ψ. In addition, a feature-wise normalization (F-norm) step is performed before calculating the MSE loss s. Method (e) [29] avoids using EMA updates of each module's weights in the second branch; however, it assigns the stop-gradient operator to the second branch to avoid collapse. Method (f) [24] has identical branches and the model optimizes the InfoNCE loss. Method (g) [27] performs an online clustering (quantization) step and the resulting code serves as a target in the loss. Methods (c) and (h) are not covered in this manuscript.

2.2 Self-supervised audiovisual learning

As we have seen in the previous section, extracting some level of supervision from data alone is essential for learning robust representations. In turn, the fundamental properties of video signals, such as their redundancy and the co-occurrence of modalities that are naturally synchronized, make them an attractive data source for self-supervised learning. In the following, based on the categorization that was introduced in Section 2.1, we review some of the most influential works on learning audiovisual representations from unlabeled videos.

2.2.1 Pretext tasks

Pretext tasks that include multi-modal inputs are significantly harder to solve, as the network has to both learn modality-specific features and also implicitly capture underlying cross-modal relations. Arandjelović and Zisserman [32], [33] formulated the Audio-Visual Correspondence task (AVC) where, given a video frame and a sound snippet, the network learns to predict whether its inputs come from the same video or not. To this end, they employ a visual and an audio sub-network for feature extraction, whereas the Euclidean distance between the resulting embeddings serves as the input for a fully-connected layer that computes the final correspondence score. Then, based on a learned threshold (the last layer's bias), the network performs a binary classification task (inputs are corresponding or not). The authors also suggest forming negative pairs of inputs, i.e. randomly sampled from different videos, to avoid collapse.

Moreover, Korbar et al. [34] proposed the Audio-Visual Temporal Synchronization task (AVTS), where the model predicts whether a video sequence and an audio sample are synchronized or not. Since this task is more difficult compared to AVC, the authors follow a curriculum learning strategy. First, the learning process starts by sampling easy negatives (mismatched audio/video pairs). Then, they progressively add harder negative pairs, i.e. excerpts from the same video that are either slightly out-of-sync or from completely non-overlapping regions of the input. They also ensure that the fraction of positive and negative pairs used during training is roughly the same, so that the network does not overfit to the pretext task. Similar to AVTS, Owens and Efros [35] designed a misalignment detection task where the corresponding video and audio inputs are artificially shifted by a few seconds. Their experiments show that this task leads to better representations, especially for classes involving human speech, compared to AVC, possibly because the solution for the latter does not require analyzing motion patterns that are present in the input video clips.

Last, Morgado et al. [36] extended this line of work to 360° videos and proposed the Audio-Visual Spatial Alignment task. Here, besides learning cross-modal correspondences, the network should also be able to reason over the entire spatial content of a 360° video by combining representations from different viewpoints that occur simultaneously. To achieve this goal, they employ a curriculum learning scheme where the model is first trained on the AVC task and then on the challenging task of spatial alignment.

2.2.2 Clustering-based methods

Following approaches based on clustering, Alwassel et al. [37] introduced Cross-Modal Deep Clustering (XDC), a framework which sets the cluster assignments of one modality (e.g. video) as targets or pseudo-labels for the opposing modality (e.g. audio) and vice

versa. In their experiments, the authors show that this technique outperforms single-modality clustering, while it also trivially avoids empty clusters which could lead to collapse. Additionally, this method is inherently scalable in terms of data used during pre-training and retains its representational power (with a small drop in performance) when combined with uncurated data. Furthermore, Asano et al. [38] extended their previous work [17] on self-labeling to videos. In fact, following the protocol of multi-modal single labeling, they achieve non-degenerate clustering via optimal transport and they also force the learned clusters to be invariant to standard augmentations, as well as the choice of modality. Moreover, prior to clustering, they introduce an alignment mechanism (in the form of a learned permutation matrix) to synchronize the outputs of the audio and video encoder, respectively. Experimental results show that the proposed method has an additional advantage of learning effective audio-visual representations that are useful for downstream tasks.

2.2.3 Contrastive learning methods

Inspired by NPID [21], Morgado et al. [39] developed the Audio-Visual Instance Discrimination task (AVID). More specifically, the authors argue that optimizing cross-modal similarity would lead to more powerful representations. Their experiments validate this hypothesis and also show that including within-modal discrimination in this setup (either as a standalone task or combined with cross-modal discrimination) in fact deteriorates performance. In addition, they observe some form of collapse in the learned representations (features are not uniformly spread in the unit sphere); thus, they decided to reinforce AVID with a calibration mechanism, called Cross-Modal Agreement (CMA), that groups together multiple videos with high similarity scores in both the video and audio feature spaces. As a result, the learning objective can be divided into two terms: AVID loss (for each query and modality, optimize a contrastive loss using a positive and a set of negatives from the opposing modality) and a combination of AVID and CMA loss (for each query and modality, first draw a positive set of samples from the same modality based on their similarity to the query in both latent spaces, and then maximize their agreement), respectively.

In a concurrent work [40], the same authors proposed an improved variant of AVID that is more robust to noisy audio-visual correspondences. Such "noise" may arise from uninformative audio-video pairs that are treated as positives (e.g. edited videos that contain background music instead of the original audio recording), as well as from semantically similar samples that are used as negatives in the contrastive objective. To mitigate the first issue, based on the fact that low similarity scores are indicative of low-quality correspondences, they formulate a weighted contrastive loss that reduces the contribution of false positives during training. In the case of false negatives, they introduce a soft target distribution that estimates the relationship between the query and each negative instance. This way, given that a negative is highly similar to the query, the loss forces their embeddings to remain close in the latent space. Experiments also demonstrated the need for an initial warmup stage using only the AVID loss, so that the network progressively learns to correct any type of weak correspondences.

Furthermore, other approaches focus on developing strong augmentation schemes to facilitate learning. Patrick et al. [41] mention that spatial cropping offers significant performance boost for downstream tasks; yet, due to high memory and processing costs, it is infeasible to use it at sufficiently large scales when applied directly to the input space.

To this end, they propose to apply multi-scale cropping at the feature level, which leads to a large number of crops per input without the need for extra memory or computational resources. They also replace the naive average pooling layer (used for aggregating information along the time axis) with a transformer-based attention module that is empirically shown to capture temporal dependencies much more effectively. Extending this line of work, Patrick et al. [42] studied the effect of composing a hierarchy of data transformations to achieve both invariance and distinctiveness[3]. In fact, following the training protocol of SimCLR [24], they show that being distinctive to temporal shifts and invariant to time reversal, as well as the choice of modality, offers a strong learning signal, while the resulting representations even surpass supervised pre-training on downstream action recognition and retrieval tasks.

2.3 SOTA on self-supervised video learning

So far, we have discussed the basics of self-supervised learning, i.e. methods found in the literature, their implementation details and also the pivotal design choices that prevent networks from learning collapsed representations during training. Additionally, we have covered some of the applications of self-supervised methods on deep audiovisual learning. However, due to the variety of existing network architectures and pre-training datasets, it is important to establish a common evaluation suite that would detach the underlying method from such choices as much as possible and enable a fair comparison. To this end, the research community agreed on using two datasets, namely UCF-101 [43] and HMDB-51 [44], as benchmarks for measuring the downstream performance of self-supervised models on the task of human action recognition.

In Table 2.1, we present the state-of-the-art of unsupervised spatiotemporal representation learning techniques on both benchmarks. For the sake of completeness, we refer to both uni-modal and multi-modal approaches; however, in both cases, evaluation on downstream datasets is conducted using only the visual modality. Moreover, since multi-modal methods require modality-specific encoders that increase memory usage, a shallow (18-layer) ResNet is commonly used as video encoder (backbone). On the contrary, when data belong to a single modality, the encoder can be shared across inputs which in turn enables the use of deeper networks (50-layer ResNet) or even more sophisticated architectures such as the Vision Transformer (ViT).

Based on Table 2.1, it is evident that the gap between supervised and self-supervised pre-training is slowly closing. In fact, the recently proposed Video Masked Autoencoder [45] yields results comparable to a supervised method that is directly optimized on the downstream classification task using a tailored architecture (TSN [46]). We also mention that the discrepancy between the results on UCF-101 and HMDB-51 is in accordance with previous studies [47] which show that the latter benefits most from motion features, i.e. optical flow, while the performance degrades when either video frames or any other modalities are involved. Furthermore, we observe that (non-) contrastive learning methods, irrespective of the input modality, are consistently among the most successful ones and only a single method that is based on clustering (XDC [37]) appears in the table. We also conclude that uni-modal approaches, perhaps surprisingly, outperform their multi-modal counterparts even when the latter are pre-trained on large databases

[3] An optimal representation should be *invariant* to transformations that do not alter the meaning of the input (e.g. cropping and geometric distortions) and *distinctive* to changes that are likely to affect its semantics (e.g. different views from different instances).

Method	Backbone	Modality	T	UCF	HMDB
VideoMoCo [49]	R(2+1)D-18	V	16	78.7	49.2
TCLR [50]	R(2+1)D-18	V	16	88.2	60.0
CVRL [51]	R3D-50	V	32	92.2	66.7
ρBYOL [48]	R(2+1)D-18	V	32	94.4	72.2
VideoMAE [45]	ViT-B	V	16	**96.1**	**73.3**
XDC [37]	R(2+1)D-18	V+A	32	84.2	47.1
RxID [40]	R(2+1)D-18	V+A	32	85.6	55.0
AVID [39]	R(2+1)D-18	V+A	32	87.5	60.8
[39] (AudioSet)	R(2+1)D-18	V+A	32	**91.5**	**64.7**
GDT [42]	R(2+1)D-18	V+A	32	89.3	60.0
Supervised [52]	TSN	V	(*)	98.6	84.3

Table 2.1: State-of-the-art results on UCF-101 and HMDB-51 using self-supervised learning. Top-1 accuracy (in %) is selected here as the evaluation metric. Unless otherwise specified, all methods are pre-trained on the large-scale Kinetics-400 dataset. V and A denote visual and audio modality, respectively. T stands for the number of frames used during fine-tuning on downstream datasets. For reference, we also mention a top-performing supervised learning method in the last row (* denotes that the entire video is passed through a frame selection mechanism which dynamically reduces the input along the time axis).

consisting of millions of samples as in the case of AudioSet[4]. For instance, ρBYOL [48], a simple method that encourages temporal persistency across features extracted from a set of ρ clips per video (the results denoted here correspond to $\rho = 4$), pre-trained on Kinetics-400 outperforms AVID [39] pre-training on the much larger AudioSet by almost 3% and 7.5% on UCF-101 and HMDB-51, respectively. Note that this comparison is not exactly fair since BYOL falls under the category of non-contrastive methods, while AVID is a contrastive learning technique; however, using BYOL or any other similar approach with audio-visual data would require storing two sets of weights (original and momentum encoder) per modality in memory, which is practically infeasible without sacrificing other vital parts of the system that also affect its performance (e.g. network's depth, batch size, number of frames per input clip, or even attention modules). Therefore, despite recent advances in the field of multi-modal self-supervised learning, it is clear that there is still much room for improvement.

[4] https://research.google.com/audioset/

Chapter 3

Methodology

In this chapter we detail the two types of learning methods used here for self-supervised pre-training. The first method involves AVID [39], an established contrastive learning technique, with a modified criterion that improves robustness against noisy pairs of positive samples. The second method is the recently proposed VICReg [31], which enables the use of multi-modal inputs due to its inherent asymmetry. Note that the latter method falls under the category of non-contrastive learning, an area that has not been yet explored for audio-visual data.

3.1 Cross-modal Instance Discrimination

Let x^i be the input video of instance i from the pre-training dataset, which can be further decomposed into a video stream and an audio stream, respectively. Through a pre-processing step, we extract a single video clip $x_v^i \in \mathbb{R}^{3\times T\times H\times W}$ and its accompanying audio in the form of a spectrogram $x_a^i \in \mathbb{R}^{F\times t}$. In the case of the video clip, T denotes the number of frames, while H and W refer to its spatial dimensions. For spectrograms, F and t denote frequency and time dimensions, respectively. Moreover, we augment each modality using transformations sampled from a distribution $\mathcal{T}$. Each input is then passed through a modality-specific sub-network, i.e. either a video encoder f_v or an audio encoder f_a, resulting in the representations $y_v^i \in \mathbb{R}^K$ and $y_a^i \in \mathbb{R}^K$, respectively. Note here that the representation space is useful for downstream tasks; however, it was empirically shown that applying the contrastive loss in a lower-dimensional space yields better representations. As a result, two projector modules h_v and h_a map each representation to the embedding space $\mathbb{R}^D$, $D < K$.

Let $v_i \in \mathbb{R}^D$ and $a_i \in \mathbb{R}^D$ be the embeddings of x_v^i and x_a^i, respectively. For each input embedding, calculating the contrastive loss requires both a positive and a set of N negative samples. Therefore, we use two types of memory banks (one per modality) to sample the desired targets on-the-fly during training. Note that each memory bank is randomly initialized and contains the entire dataset's embeddings from the last training iteration. Furthermore, based on the definition of the Instance Discrimination task, each sample $j \neq i$ from the dataset can be treated as a negative, thus we randomly select N embeddings from the rest of the dataset to form the set of negative targets for instance i. The last step involves mapping the embeddings to the D-dimensional unit sphere (via L2 normalization). This way, since the embeddings become unit vectors, the agreement score between each pair of embeddings is equivalent to their cosine similarity and does not depend on their scale. In Figure 3.1 we present an overview of the proposed method.

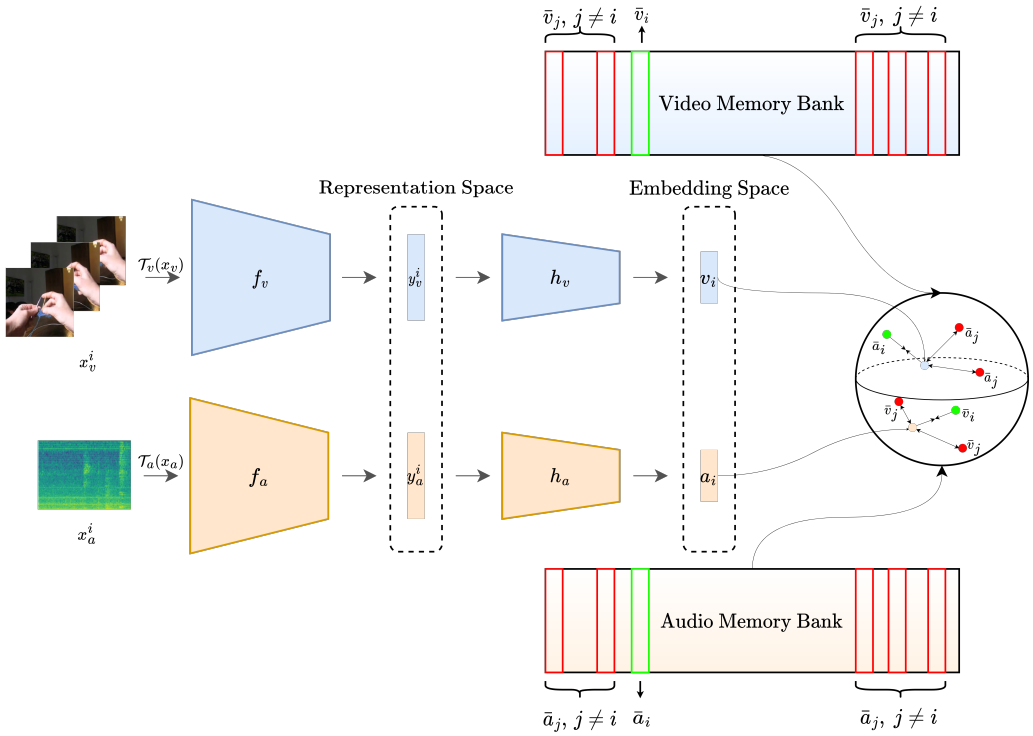

Figure 3.1: Cross-modal Instance Discrimination (xID) framework

Now we have to define the training objective. Let x_i be the embedding of instance i and $\bar{y} = (\bar{y}_i, \{\bar{y}_j\}_{j=1}^N)$ its associated targets sampled from the memory bank. A standard choice for contrastive learning involves the InfoNCE loss

$$\mathcal{L}_{InfoNCE}(x_i, \bar{y}) = -\log \frac{\exp(x_i^T \bar{y}_i/\tau)}{\exp(x_i^T \bar{y}_i/\tau) + \sum_{j=1}^N \exp(x_i^T \bar{y}_j/\tau)} \tag{3.1}$$

where τ is a temperature hyperparameter that controls the softness of the output distribution. In this context, it is typical to set $\tau << 1$ to avoid gradient saturation (the partial derivative of $\mathcal{L}_{InfoNCE}$ with respect to its input x_i is inversely proportional to τ).

Although this loss works reasonably well for most applications, it does not address the common issue of weak correspondences between audio-visual data which, in turn, leads to low similarity scores $x_i^T \bar{y}_i$. In fact, the InfoNCE loss assigns stronger gradients to those noisy inputs (to maximize their agreement), resulting in a corrupted training signal. By drawing connections to supervised binary classification with noisy labels, Chuang et al. [53] proposed the Robust InfoNCE (RINCE) loss

$$\mathcal{L}_{RINCE}(x_i, \bar{y}) = -\frac{\exp(q \cdot x_i^T \bar{y}_i/\tau)}{q} + \frac{\lambda^q \cdot [\exp(x_i^T \bar{y}_i/\tau) + \sum_{j=1}^N \exp(x_i^T \bar{y}_j/\tau)]^q}{q} \tag{3.2}$$

where q, $\lambda \in (0, 1]$ are hyperparameters that control the level of robustness against noisy inputs and the contribution of negatives, respectively. It is clear that reducing λ places more emphasis on the positive score, whereas for $\lambda \to 0$ the objective becomes non-contrastive, i.e. free of negative pairs. In addition, for $q \to 0$, RINCE is asymptotically equivalent to the InfoNCE loss, while for $q \to 1$ we get its fully symmetric form that is necessary for a noise-tolerant loss function.

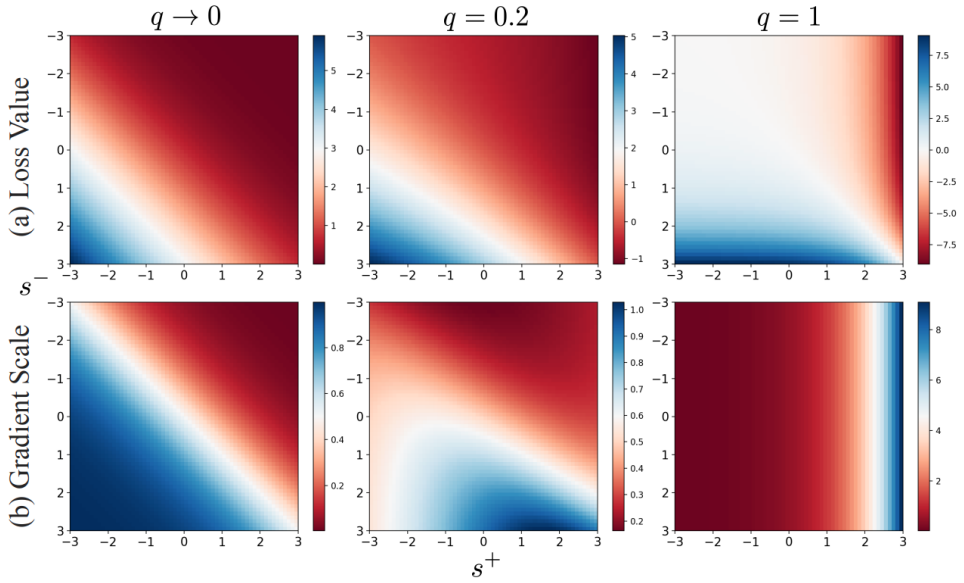

Figure 3.2: (taken from [53]) Visualization of the Robust InfoNCE (RINCE) loss landscape and gradient scale with respect to the positive score $s^+ = x_i^T \bar{y}_i / \tau$ and the negative scores $s^- = x_i^T \bar{y}_j / \tau$. Here, λ is fixed to 0.5, while q takes three different values (close to 0, 0.2 and 1).

Figure 3.2 depicts the evolution of RINCE loss in terms of its value and gradient scale for different values of q. In the case of InfoNCE loss (first column), we observe that the strongest gradient signals are derived from pairs with either low positive scores or high negative scores. As q increases (second column), the loss places more weight on pairs with relatively high positive and negative scores, leading to the edge case of $q = 1$ (third column) where training solely depends on positive pairs of samples with high scores. An alternative interpretation of RINCE loss is that it performs a filtering of positive pairs, with parameter q resembling the filter's cutoff. As a result, there appears to be a trade-off between faster convergence ($q \to 0$) and robustness ($q \to 1$). In practice, it is recommended to set $q \in [0.1, 0.5]$ range, starting from a low value and linearly increasing it during training. This way, the entire underlying space is being actively explored in the first training steps, whereas the loss progressively discards noisy inputs by assigning small gradients to uninformative positive pairs.

In our case, we formulate the contrastive objective as follows: Let $\bar{v} = (\bar{v}_i, \{\bar{v}_j\}_{j=1}^N)$ and $\bar{a} = (\bar{a}_i, \{\bar{a}_j\}_{j=1}^N)$ be the targets for the embeddings (v_i, a_i). Following [39], we opt for cross-modal discrimination since it provides the best results compared to other forms of this task (i.e. within-modal or joint discrimination). Using Equation 3.2, we define the cross-modal instance discrimination (xID) loss for instance i

$$\mathcal{L}_{xID} = \frac{\mathcal{L}_{RINCE}(v_i, \bar{a}) + \mathcal{L}_{RINCE}(a_i, \bar{v})}{2} \tag{3.3}$$

It is worth mentioning that all contrastive objectives, including RINCE, are not robust to false negatives, i.e. samples in the negative set that are semantically similar to the input (thus having high negative scores s^-). In fact, as can be seen in Figure 3.2, hard negative mining yields a significant gradient signal that facilitates the learning process. Therefore, to avoid this issue, it is reasonable to choose a non-contrastive method for self-supervised pre-training.

3.2 Variance-Invariance-Covariance Regularization

The success of contrastive learning methods lies primarily in negative sampling, which prevents the network from collapsing, i.e. mapping arbitrary inputs to constant representations. On the contrary, non-contrastive techniques are based on asymmetric architectures with shared weights, where the second branch is not updated during backpropagation, to avoid collapse. VICReg (short for Variance-Invariance-Covariance Regularization) [31], which belongs to the latter category, addresses the issue of collapse through a novel loss function that is applied to each branch independently. As a result, since the choice of network architecture is detached from the learning method, VICReg is eligible for multimodal data as well (where a branch with a dedicated set of weights is required per input modality).

More specifically, the authors of [31] mention two types of collapse that typically occur in a self-supervised setting. The first type involves trivial solutions, i.e. the network optimizes the objective without inducing an informative latent space. The second type, which is called informational collapse, arises from highly correlated embedding variables (the network encodes similar information to different embedding dimensions). To circumvent these problems, they propose a training objective consisting of three regularization terms that are necessary for stabilizing the training process.

Let $Z = [z_1, ..., z_n]$ and $Z' = [z'_1, ..., z'_n]$ be a batch of n embeddings (each of dimensionality d) from the first and second network's branch, respectively. The invariance term is defined as the mean-squared distance between each pair of embeddings

$$s(Z, Z') = \frac{1}{n} \sum_{i=1}^{n} \|z_i - z'_i\|_2^2 \tag{3.4}$$

The variance term is a hinge loss that maintains the standard deviation of each embedding dimension above a predefined threshold. Concretely, by forcing the embeddings within a batch to be different, this function prevents the entire system from performing a simple constant mapping. Denoting $z^j = [z_1^j, ..., z_n^j]$ as the vector of values from the j-th dimension of each embedding in batch Z, the variance term is calculated as follows

$$v(Z) = \frac{1}{d} \sum_{j=1}^{d} \max\left(0, \gamma - \sqrt{\mathrm{Var}(z^j) + \epsilon}\right) \tag{3.5}$$

where γ, ϵ hyperparameters that refer to the standard deviation threshold (here it is fixed to $\gamma = 1$) and a small scalar to prevent any numerical instability, respectively.

The last term, called covariance loss, decorrelates the different embedding dimensions by pushing the off-diagonal elements of the covariance matrix towards zero. In particular, the $d \times d$ covariance matrix for batch Z is defined as

$$C(Z) = \frac{1}{n-1} \sum_{i=1}^{n} (z_i - \bar{z})(z_i - \bar{z})^T$$

where $\bar{z}$ is the embeddings' mean and the covariance loss term becomes

$$c(Z) = \frac{1}{d} \sum_{i \neq j} [C(Z)]_{i,j}^2 \tag{3.6}$$

From Equations 3.4, 3.5 and 3.6, we derive the final training criterion as follows

$$\ell(Z, Z') = \lambda_{inv} s(Z, Z') + \lambda_{cov} \left[c(Z) + c(Z')\right] + \lambda_{var} \left[v(Z) + v(Z')\right] \tag{3.7}$$

where λ_{inv}, λ_{cov}, λ_{var} hyperparameters that weigh the importance of each term. Note that the choice of these hyperparameters is crucial, since any setup except for $\lambda_{cov} = 1$ and $\lambda_{inv} = \lambda_{var} > 1$ leads to unstable training.

In our case, the system is similar to that of Figure 3.1 with the following modifications. First, each memory bank is discarded since VICReg does not require negative samples. Second, the projector modules are replaced by the so called expander modules, which map the encoders' outputs to a higher dimensional embedding space $\mathbb{R}^D$. The choice of $D > K$ is empirically justified, since feature decorrelation performs better on higher dimensions. Third, the embeddings are not L2 normalized, as it was found to deteriorate the representations' quality in this setting. Denoting $V = [v_1, ..., v_n]$ and $A = [a_1, ..., a_n]$ a batch of n video and audio embeddings, respectively, the VICReg loss for audio-visual pre-training is equivalent to that of Equation 3.7

$$\mathcal{L}_{VICReg} = \lambda_{inv} s(V, A) + \lambda_{cov} \left[c(V) + c(A)\right] + \lambda_{var} \left[v(V) + v(A)\right] \tag{3.8}$$

and in our experiments, we use the coefficient values suggested in [31], i.e. $\lambda_{cov} = 1$, $\lambda_{inv} = \lambda_{var} = 25$.

In Chapter 4, we also consider the case of uni-modal (video-only) VICReg as a baseline. In this scenario, the architecture becomes symmetric, i.e. both the visual encoder's and the expander's weights are shared across branches. Moreover, the training criterion is identical to that of Equation 3.8 with the batch of audio embeddings A being replaced by $V' = [v'_1, ..., v'_n]$, where v'_i refers to the expander's output for an augmented version of video clip i.

Chapter 4

Experimental evaluation

In this chapter, we focus on both the quantitative and qualitative analysis of audio-visual self-supervised learning methods by conducting a series of experiments. First, in Section 4.2 we assess the performance of all models on standard benchmarks by means of linear classification and network fine-tuning. In fact, the former approach is useful for testing linear separability in the resulting feature space, whereas the latter simulates a transfer learning scenario. Next, in Section 4.3 we consider the task of measuring the models' generalization ability. To this end, each downstream dataset is first split into two distinct categories (*seen* and *unseen* concepts) based on the bias introduced by the pre-training dataset, and then a linear classifier is trained per split independently. Last, any information regarding our implementation can be found in Section 4.1.

Throughout this chapter, we refer to all learning methods using the following abbreviations

- *xID*: Cross-modal instance discrimination task. For each instance i, the number of negative targets per modality is set to $N = 1024$. At the end of each iteration, the targets $\bar{x}_i$ are refreshed based on an exponential moving average (EMA) update $\bar{x}_i \leftarrow m \cdot \bar{x}_i + (1 - m) \cdot x_i$, where x denotes either visual or audio modality, x_i is the up-to-date embedding and $m = 0.5$. Furthermore, similarity scores are scaled using a constant hyperparameter (temperature) $\tau = 0.07$. This prevents vanishing gradients, since $\frac{\partial \mathcal{L}_{xID}}{\partial x_i} \propto \frac{1}{\tau}$, and facilitates training. For RINCE loss, we set $\lambda = 0.01$, while q linearly increases from 10^{-3} to 0.4 throughout training.
- *VICReg-AV*: Multi-modal VICReg.
- *VICReg-V*: Uni-modal VICReg that is solely trained on video frames.
- *Supervised*: Supervised pre-training baseline, where video and audio representations are concatenated prior to classification, used to roughly estimate an upper bound on the expected downstream performance.
- *Random*: Here, network weights are randomly initialized (no pre-training involved), indicating a lower bound on performance.

4.1 Implementation details

In the following paragraphs, we detail all aspects of both training and downstream evaluation, including datasets, pre-processing pipelines, network architectures and optimization. Note that these settings are identical for all learning methods to ensure a fair comparison.

4.1.1 Datasets

VGGSound

VGGSound [54] is a large-scale dataset collected from YouTube that guarantees audio-visual correspondence, i.e. sound sources are visually evident, and low label noise. All videos have constant length (10 seconds) and depict real-world scenes or actions. Upon its release, it consisted of more than 200,000 videos spanning 310 classes. However, possibly due to content or accounts being deleted by users, the total number of videos is currently close to 170,000 and the number of classes has been reduced to 309. These classes cover a wide range of applications such as human actions, sports events, playing musical instruments, or even miscellaneous activities involving animals, natural phenomena and machinery. Note that all scenes may appear either in domestic environments or outdoors, resulting in high video content variations.

Furthermore, the use of in-the-wild videos creates a rather challenging learning scenario caused by weak correspondences (false positives). For example, it is common for users to upload edited videos, where the original audio recording is replaced by background music. In addition, there also exist hundreds of videos in this dataset that lack an audio stream[1]. There are two possible explanations for this: either the original uploads involved silent videos or their accompanying audio is no longer accessible (could have been removed by the platform due to copyright restrictions or the download links for those audio tracks are now locked). This, however, does not greatly affect our study; in fact, it is certain that weakly correspondent pairs of audio and video occur in the training set.

For our experiments, we only consider a subset of the original training set that consists of 50,000 videos across all 309 classes. The process of collecting this subset is the following: First, we extract a balanced subset using the minimum number of samples per class as a threshold, leading to approximately 33,500 videos. Then, the desirable dataset length is reached by randomly sampling data from the remainder of the original set. It is also worth mentioning that there are more than 200 videos in our training subset that do not have an audio stream.

This subset is only used here for pre-training, i.e. we do not intend to evaluate our models on the official test set as VGGSound is tailored for the task of audio recognition. Moreover, for self-supervised methods, all label information is discarded. Last, during supervised pre-training, we use a disjoint set of 10,000 samples from the original training set for validation (to prevent overfitting).

UCF-101

UCF-101 [43] is an action recognition dataset consisting of 13,320 in-the-wild videos in total (collected from YouTube). It offers a diverse set of human actions that take place in realistic settings. As a result, data exhibit large variations in viewpoint, background, object pose and scale, lighting conditions and camera motion. Moreover, the 101 classes can be organized into 5 types (human-object interaction, body-motion only, human-human interaction, playing musical instruments, sports) and 25 groups, where the entities (4-7 videos) of each group may share common attributes such as matching background or

[1] An example that belongs to our training subset is available here: https://www.youtube.com/watch?v=q3Wl0maNldU

viewpoint. Thus, to prevent data leakage, it is important to keep videos from the same group either in training or test set.

All videos have a fixed frame rate (25 fps) and a resolution of 320 × 240 pixels (240p class). Also, the average clip length is 7.21 seconds (minimum is 1.06 seconds, whereas the maximum is 71.04 seconds). From the total of 101 classes, only 51 contain audio. This does not affect evaluation, since only the visual modality is considered for downstream tasks; however, this subset of 51 classes is useful for all retrieval experiments as detailed in Appendix A.

There exist three official train/test splits for UCF-101. However, given the amount of time consuming experiments conducted in this study, we only report results on *Fold 1*. The total number of training data is equal to 9,537 (ranging from 72 to 121 samples per class), whereas the number of videos in test set is 3,783 (ranging from 28 to 49 samples per class).

HMDB-51

HMDB-51 [44] is another popular benchmark that is suitable for the task of action recognition. The main difference here is that most videos are collected from movies, whereas only a small percentage of data is gathered from public databases. The 51 action classes can be divided into 5 types, namely facial expressions (e.g. smile, talk), facial actions with object manipulation (e.g. eat, drink), general body movements (e.g. jump, handstand), body movements with object interaction (e.g. shoot ball, brush hair) and also for human interaction (e.g. hug, fencing), respectively.

As in the case of UCF-101, we use Fold 1 (out of the three official train/test splits) for evaluation. This subset is fully balanced, since there are 70 videos per class for training (3570 samples in total) and 30 videos per class for testing (a total of 1530 samples). Regarding video clips, their duration ranges from just below 1 second to over 5 seconds, while the overall video quality can be either high (all visual elements are clearly depicted), medium (only large body parts are identifiable) or low (large amount of motion blur and compression artifacts). Moreover, the creators of HMDB-51 applied a normalization step that involves converting the frame rate to 30 fps and scaling the height of all frames to 240 pixels (the width was scaled accordingly to maintain the original aspect ratio).

4.1.2 Architectures

Video Encoder

As the video feature extractor, we use a ResNet architecture with 18 layers that performs separate spatial and temporal convolutions on each input volume, denoted as R(2+1)D-18 [55]. In fact, the authors of [55] empirically show that this spatiotemporal factorization leads to faster convergence compared to networks that consist of 3D convolutional blocks, especially as depth is increased. An additional advantage of R(2+1)-D networks is that they outperform their R3D counterpart by a significant margin, leading to approximately 3.5% higher accuracy with the same computational cost (FLOPs).

In Table 4.1 we present an overview of the video encoder's architecture. The first row stands for the input video clip, whereas the last layer's (max pooling) output forms the 512-dimensional feature vector (representation) for this input. Also, note that each residual block contains 2 sets of weights per convolution type, i.e. spatial and temporal, and each convolution is followed by batch normalization and ReLU activation, respectively.

Table 4.1: Video encoder architecture. Each block's input passes through 4 convolutional layers (2 spatial and 2 temporal), and each of those layers is followed by batch normalization and ReLU activation. X_s and X_t denote the output's spatial and temporal dimensions, respectively, and C is the number of output channels. K_s and S_s stand for spatial kernel size and stride, respectively, while the same parameters for temporal convolutions are represented by K_t and S_t. All convolutions are padded accordingly to maintain the original input dimensions (any dimensionality reduction occurs only through strides > 1 or max pooling layers).

Layer	X_s	X_t	C	K_s	K_t	S_s	S_t
video	224	8	3	-	-	-	-
conv1	112	8	64	7	3	2	1
max-pool	56	8	64	3	1	2	1
block2 ($\times 2$)	56	8	64	3	3	1	1
block3.1	28	4	128	3	3	2	2
block3.2	28	4	128	3	3	1	1
block4.1	14	2	256	3	3	2	2
block4.2	14	2	256	3	3	1	1
block5.1	7	1	512	3	3	2	2
block5.2	7	1	512	3	3	1	1
max-pool	1	1	512	7	1	1	1

Audio Encoder

Following [39], the audio backbone is a VGG-style 2D convolutional network. Here, a block is defined as a sequence of a convolutional layer, batch normalization and ReLU activation (no skip connections involved). The overall architecture is depicted in Table 4.2 for an input spectrogram that corresponds to 1 second of audio.

Table 4.2: Audio encoder architecture. X_f and X_t denote the output's frequency and time dimensions, respectively, and C is the number of output channels. K_f and S_f are the kernel size and stride for the frequency axis, respectively, and the same parameters for time axis are denoted as K_t and S_t.

Layer	X_f	X_t	C	K_f	K_t	S_f	S_t
audio	128	100	1	-	-	-	-
conv1	64	50	64	7	7	2	2
block2.1	32	25	64	3	3	2	2
block2.2	32	25	64	3	3	1	1
block3.1	16	13	128	3	3	2	2
block3.2	16	13	128	3	3	1	1
block4.1	8	7	256	3	3	2	2
block4.2	8	7	256	3	3	1	1
block5.1	8	7	512	3	3	1	1
block5.2	8	7	512	3	3	1	1
max-pool	1	1	512	8	7	1	1

Projector module

For *xID* method, we denote the sub-network that maps features to the embedding space $\mathbb{R}^D$ as the projector. This module is a simple multi-layer perceptron (MLP) composed of three fully connected layers, each one followed by ReLU activation. The hidden dimension of the first two layers is equal to 512 (same as the feature vector's length), while the last layer has 128 hidden units in total. Therefore, the dimensionality of the embedding space is $D = 128$, i.e. $4\times$ smaller than the representation space.

Expander module

The expander is useful for all VICReg-based methods. As in the case of the projector, it is a three-layer MLP (all hidden dimensions are set to 4096), with intermediate batch normalization and ReLU activation layers. The resulting embeddings are 4096-dimensional, i.e. $8\times$ larger than the original representations. It is worth mentioning that the authors of [31] showed that even higher dimensional embedding spaces lead to additional performance gains; however, as this comes at the expense of large GPU memory consumption, we use $D = 4096$ since it prevents us from sacrificing other aspects of the training process.

4.1.3 Training setup

All systems are implemented in PyTorch [56] and trained on GRNET's high-performance computing cluster ARIS[2] using data parallelism. More specifically, we use the distributed data parallel strategy that allocates a copy of the model weights on all devices, computes the gradient for disjoint batches and then synchronizes the individual gradients with an all-reduce algorithm (so that all copies are equally updated). In our case, the batch size per worker is fixed to 16. Moreover, all self-supervised models are pre-trained for 100 epochs. Thus, using 8 Nvidia Tesla K40m (the effective batch size is equal to $16 \cdot 8 = 128$), training takes approximately 11 ½ and 13 ½ days for *xID* and all VICReg based methods, respectively.

For the *Supervised* baseline, we attempt to solve a classification task using the original labels from VGGSound dataset. As we expect the model to overfit rather quickly on the task (our subset is $\sim 3.5\times$ smaller than the official training set), we initially fix the number of pre-training epochs to 50 and monitor accuracy on a disjoint validation set consisting of 10,000 samples. However, training was early stopped after epoch 12 due to stagnant validation accuracy during the last two epochs.

Next, we present additional information about the pre-processing pipeline, involving clip extraction and augmentation per input modality, and also the process of optimization. Note that those steps are shared across methods during pre-training.

Video pre-processing

Each input clip consists of $T = 16$ frames, corresponding to 1 second of video extracted at a frame rate of 16 fps. Moreover, to improve data efficiency, we randomly sample a total of 5 clips from each video per epoch, i.e. the effective dataset length is $50,000 \cdot 5 = 250,000$ samples. For *VICReg-V* method, given the fact that only the visual modality is available for pre-training, we form a positive pair by drawing a second clip that is d seconds apart

[2] https://hpc.grnet.gr/

from the first clip (anchor). The parameter $d \in [2, 8]$ range is randomly sampled during training and is used to enforce temporal persistency on the resulting representations [48].

Next, we apply a set of transformations on each input clip (also known as data augmentation) to enforce invariance against nuisance factors, such as color patterns or local geometric deformations. To this end, following [39], we use three types of augmentations, namely multi-scale cropping with 8% minimum area, random horizontal flipping with probability 0.5 and color jittering.

The first type involves extracting crops using a portion of each frame (controlled by the scale parameter that is randomly sampled from $[0.08, 1]$ range) with a randomly selected aspect ratio (in $[3/4, 4/3]$ range). Then, to ensure that the outputs have similar spatial dimensions, all crops are resized to a shape of 224×224 using bi-linear interpolation.

Color jittering refers to changes in brightness, contrast, saturation and hue. Here, we choose to jitter the first three properties by a factor of 40% (e.g. the final brightness will be somewhere between [60%, 140%] of the original value), whereas hue's factor is set to 20%, i.e. output hue is within [-20%, 20%] range of the input value.

Last, we perform a normalization step on all inputs (up until this point, each frame consists of 8-bit unsigned integer values). Following standard protocol, we first map all clips to [0, 1] range and then normalize using the mean and standard deviation derived from ImageNet.

Audio pre-processing

For each video clip, we extract an accompanying spectrogram corresponding to 2 seconds of audio (the offset between audio's and video's starting point is randomly sampled within ± 0.5 seconds). In the following, we discuss the process of extracting spectrograms, as well as augmentation techniques, in further detail.

First, all audio waveforms are resampled to 16 kHz (common signals such as human speech or environmental sounds lack high frequency content) and converted to mono. In the case of clips with missing audio, we use a zero-filled tensor as their corresponding waveform (zero inputs do not affect model's weights during training). Then, we normalize each waveform, such that their ℓ_∞ norm is equal to 1, and decrease their amplitude by a random percentage in [0%, 30%] range.

The mapping to time-frequency domain is performed via short-time Fourier transform (STFT) that calculates DFT over short overlapping windows. The length of the windowed signal is set to 512 samples, which corresponds to 32 milliseconds of audio at a sample rate of 16 kHz, and the hop length to 256 samples (16 ms). Next, we transform frequencies from Hz to Mel scale with a total of 128 Mel bands. Thus, the reduced representation has a shape of $128 \times 62.5t$[3], with t being the audio clip's duration in seconds (here $t = 2$). Last, we augment all spectrograms by randomly applying variable-length masks that may cover up to 20% of the time and frequency axis, respectively.

Optimization

All models are trained using the Adam optimizer [57]. The initial learning rate is set to $5 \cdot 10^{-4}$ and decays by a factor of 0.9 every 10 epochs. In addition, we apply weight decay (i.e. an additional regularization term that involves the ℓ_2 norm of the model's weights) with a parameter of 10^{-5}.

[3] In our setup, 1 second of audio is split into 16,000/256=62.5 time bands. As a result, for odd values of t, the actual length of the time axis is $\lfloor 62.5t \rfloor$.

4.2 Downstream performance

4.2.1 Linear Classification

The first experiment involves linear probing, i.e. we evaluate the extent to which representations are linearly separable. To this end, we add a linear layer on top of the video encoder that maps feature vectors to class logits and solve a classification task using the common cross-entropy loss. Note that, following standard practice, only the linear layer's weights are trainable in this scenario.

Here, 0.5 seconds long video clips are extracted at a frame rate of 16 fps ($T = 8$). We do not apply any augmentations except for resizing all frames to 256×256 dimensions and then extracting a 224×224 random crop per frame. Moreover, all models are trained for 50 epochs using the Adam optimizer with a learning rate of 10^{-4} that decays by a factor of 0.3 after epoch 20, 30 and 40, respectively. Last, throughout this chapter, we report two types of results for both datasets, namely

- Clip-level results, where the network is tasked to classify each clip using the corresponding video's ground truth label. In this case, a total of 25 clips is extracted from each video.
- Video-level results, where we average the network's predictions over 10 uniformly sampled clips per video to obtain the final predicted class for the entire video.

The final results are presented in Table 4.3 and thoroughly reviewed in Section 4.2.3.

Table 4.3: Linear classification results. The reported metrics are top-1 and top-5 accuracy, respectively (in %).

Method	**UCF-101**				**HMDB-51**			
	Clip-level		**Video-level**		**Clip-level**		**Video-level**	
	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
Random	4.76	15.51	7.08	21.49	4.32	17.43	4.72	20.14
Supervised	38.37	69.11	41.05	72.10	21.05	52.45	22.31	54.59
xID	**47.25**	**77.11**	**51.20**	**80.91**	**25.75**	**58.78**	**28.08**	**61.29**
VICReg-AV	36.80	68.20	39.75	71.3	20.21	50.23	21.85	52.69
VICReg-V	46.04	75.28	49.31	77.93	21.72	54.06	23.69	56.10

4.2.2 Fine-tuning

In addition, we test the transferability of the resulting representations via fine-tuning on each downstream dataset independently. As in the case of linear classification, we initialize the network with the pre-trained video encoder's weights and append a single linear layer that outputs class logits. However, for this experiment, all weights are trainable.

Video clip duration is fixed to 0.5 seconds (a total of $T = 8$ frames per clip). Moreover, all clips are augmented during training using the same transformations as in pre-training stage, i.e. multi-scale cropping, color jittering and random horizontal flipping. Following [39], we use the same augmentation parameters as detailed in Section 4.1.3 for UCF-101; however, in the case of HMDB-51, color channels are jittered using a factor of 100% for saturation, contrast and brightness, respectively (hue factor remains at 20%).

For UCF-101 dataset, we fine-tune all pre-trained models for 16 epochs using Adam with a learning rate of 10^{-4} that decays by a factor of 0.3 after epoch 6, 10 and 14, respectively. On the contrary, training on HMDB-51 takes 10 epochs in total using the same parameters for the optimizer (here, learning rate decays after epoch 3, 6 and 8).

For *Random* method, since it does not involve any type of pre-training, we train for a total of 100 epochs regardless of the downstream dataset using the same base learning rate (10^{-4}) that decays after epoch 30, 60 and 80 by the same factor (0.3).

Prior to fine-tuning, we follow a warm-up strategy and train the output layer weights for a total of 5 epochs while keeping the encoder's weights frozen. Last, we also apply dropout with probability 0.5 to the encoder's output (512-dimensional feature vector).

Final results are shown in Table 4.4. Furthermore, we provide an in-depth analysis of our findings in Section 4.2.3.

Table 4.4: Full network fine-tuning results. Top-1 and top-5 accuracy are used here as the reported metrics.

Method	**UCF-101**				**HMDB-51**			
	Clip-level		**Video-level**		**Clip-level**		**Video-level**	
	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
Random	57.84	83.53	62.78	86.87	25.32	60.24	27.56	65.35
Supervised	57.34	84.06	63.10	87.06	32.41	65.71	37.20	69.62
xID	**66.71**	**89.31**	**73.22**	**92.78**	**38.20**	**69.60**	**42.85**	**73.69**
VICReg-AV	53.76	82.51	59.53	85.94	30.41	64.04	34.65	68.96
VICReg-V	65.39	87.22	69.46	89.48	34.10	67.90	37.07	71.52

4.2.3 Discussion

In this section, we attempt to interpret our findings in a way that would facilitate future development of audio-visual self-supervised learning methods. Before that, it is important to state the following:

- Based on previous works, it is evident that self-supervision has scaling properties, i.e. large-scale data, deeper networks and longer training time leads to downstream performance improvements. However, as our main goal is to compare methods on the same basis, we traded-off these aspects in a way that would allow us to train networks in a reasonable time using commodity hardware.

- Following from the previous point, we observed that VICReg-based methods converged slower than *xID*; thus, these would benefit more from longer pre-training. In fact, we noticed that the covariance term of Equation 3.6 decreased rapidly for the first few epochs, while the other two terms barely dropped. Then, for the rest of the pre-training stage, both invariance and variance terms were decreasing while the covariance loss kept on increasing. This behavior also occured for larger learning rates, suggesting that VICReg loss optimization is difficult in general. Moreover, it could also indicate the importance of negative targets used in *xID* method, which yield a significant gradient signal that facilitates the learning process.

Based on the results shown in Tables 4.3 and 4.4, we deduce that *xID* clearly outperforms all other methods on both benchmarks, with *VICReg-V* being a close second. Specifically, their difference in both linear probing and fine-tuning protocol does not exceed 3.5% on UCF-101, whereas for HMDB-51 it reaches up to 5.8%. In addition, it is clear that the supervised baseline does not perform as well as we might have expected, which highlights the need for eliminating weak audio-visual correspondences during pre-training; however, fine-tuning on HMDB-51 yields comparable results to *VICReg-V* model.

Furthermore, we notice that *VICReg-AV* model performs poorly on all benchmarks; for instance, transfer learning on UCF-101 yields slightly worse results compared to random initialization. We hypothesize that this performance degradation can be attributed to the following two factors. First, as already mentioned, robustness against uninformative video-audio pairs is crucial for learning high-quality representations. This could be achieved by substituting the MSE loss term with a robust regression loss, e.g. Huber loss[4] (which introduces a tunable hyperparameter that controls robustness). The second issue arises from the different learning dynamics for audio and visual modalities. Concretely, according to a study that introduced audio-visual slow-fast networks [58], the audio pathway overfits faster compared to its video counterpart, an effect that hinders training. In the case of *VICReg-AV*, since both streams interact via the invariance loss, it is clear that the video pathway will be negatively affected if the audio network stagnates (i.e. audio embeddings do not change significantly). Moreover, this effect is mitigated during *xID* pre-training as up-to-date embeddings are replaced by memory targets, suggesting that sampling from a memory bank acts as a form of *regularization* in this case. Therefore, a potential improvement for *VICReg-AV* method would be to randomly drop the audio pathway, i.e. avoid updating its weights at random intervals, to slow down its learning dynamics and make it more compatible with the visual stream.

4.3 Concept Generalization

The ultimate goal of self-supervised learning is to extract robust, high-level features from unlabelled data that can be transferred across a variety of datasets and tasks with little to no further training involved. Therefore, it is necessary to design an evaluation suite that aims to quantify the generalization performance of self-supervised models, i.e. their ability to adapt to unknown domains beyond the distribution defined by the pre-training dataset. Although the majority of methods relies on the popular benchmark discussed in Section 4.2, here we explore an alternative approach that draws connections between the underlying categories in the pre-training and target (downstream) datasets, respectively. Note that this type of benchmark already exists for visual representation learning methods [59]; yet, this is not the case with spatiotemporal data.

In Section 4.3.1 we detail the experimental procedure, while our empirical results are presented in Section 4.3.2 and discussed in further detail in Section 4.3.3.

4.3.1 Experimental setup

Although labels are not required during the self-supervised learning phase, it is clear that this information is not entirely discarded. In fact, regardless of the method that is used, the network processes data which belong to a finite set of semantic categories. It is also

[4] https://en.wikipedia.org/wiki/Huber_loss

possible that a high degree of overlap occurs between categories in the pre-training and target dataset, respectively. Thus, in an attempt to reveal those underlying relationships, we leverage a mechanism that maps class names to a common space where it is feasible to measure the semantic distance between categories.

To accomplish the aforementioned goal, we use two pre-trained language models found in HuggingFace repository, namely `all-MiniLM-L6-v2`[5] and `phrase-bert`[6]. Each model maps pre-processed class names (i.e. after tokenization and punctuation removal) to a low-dimensional dense vector space. Then, using cosine similarity as the predefined metric, we find a total of 3 classes per downstream dataset (UCF-101 and HMDB-51) that are most similar to each class of the pre-training dataset. Note that the choice of language models is arbitrary and based on the fact that they yield a single vector for inputs consisting of multiple words (another option would be averaging word-level embeddings). As a result, this experiment would work with any model that produces dense embeddings from natural language inputs.

The last step involves filtering the results since they contain a lot of false positives. To consider a group of classes semantically similar, the following criteria should be fulfilled:

- The similarity measure between the query class (i.e. from the pre-training dataset) and the top-1 predicted class (i.e. from each target dataset) of `all-MiniLM-L6-v2` model should exceed a predefined threshold.
- For each query class, the intersection between the top-3 predictions of both models should not be an empty set.

Let us now elucidate why we chose these criteria. First, given the fact that the former model (`all-MiniLM-L6-v2`) is trained on 3× more data than `phrase-bert`, we consider it the most powerful. We also observed that this language model produces slightly more accurate matches than `phrase-bert`. For example, in the case of a true positive pair, the corresponding target class is usually the top-1 ranked prediction of `all-MiniLM-L6-v2`, whereas for `phrase-bert` it can be found among the top-3 predictions. This motivates the second criterion, i.e. a group of similar classes is considered true positive when both models reach a consensus. Furthermore, to eliminate false positives, we empirically set a threshold for the cosine similarity metric. More specifically, we found that values in the range of $[0.45, 0.50]$ work well in practice and choose 0.47 as the threshold for this setup (we observed that values close to 0.5 might eliminate a few true positives). Last, upon further inspection, we manually filter some detected outliers, i.e. false positives (this step is optional).

Based on the outcome of this experiment, we split the categories in each downstream dataset into two groups, namely *seen concepts*, i.e. classes that are related to one or more categories existing in the pre-training dataset, and a disjoint set denoted as *unseen concepts*. Table 4.5 illustrates the number of classes that constitute the collection of seen concepts per downstream dataset. For this analysis, we also use categories found in other popular pre-training datasets such as Kinetics-400 and AudioSet.
These, perhaps surprising, results show that the most commonly used dataset (Kinetics-400) has more than 80% class overlap with this particular benchmark suite, thus it does not provide useful insights into generalization performance. Conversely, for the other two pre-training datasets, this overlap does not exceed 55% in any case.

[5] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2
[6] https://huggingface.co/whaleloops/phrase-bert

Table 4.5: Cardinality of *seen concepts* set for different pre-training datasets. There is a total of 309, 527 and 400 classes in VGGSound, AudioSet and Kinetics-400, respectively.

Pre-training dataset	UCF-101	HMDB-51
VGGSound	54/101	22/51
AudioSet	50/101	28/51
Kinetics-400	86/101	41/51

4.3.2 Results

Following the formation of *seen* and *unseen concepts* per downstream dataset, we need to evaluate performance on each set independently. To this end, based on the experimental protocol introduced in Section 4.2.1, we first partition each dataset into these two parts and train a single linear layer per part on features extracted from the (shared) pre-trained video encoder. Moreover, to test class-level feature alignment, we use a varying number of training data per class (in all cases, the test set remains as is).

Final results on UCF-101 are shown in Figures 4.1 and 4.2, whereas Figures 4.3 and 4.4 refer to HMDB-51 dataset.

4.3.3 Discussion

Measuring generalization performance using this setup is unarguably a challenging task. For example, we extract both sets of seen and unseen concepts based on the assumption that class names provide a rich description of the underlying data semantics. However, this does not necessarily hold for vague descriptions (e.g. most classes in HMDB-51 are represented by single words) or lexically ambiguous words (e.g. names such as "dribble" or "smoke" can be found in the list of HMDB-51 classes). In addition, another issue arises from imbalanced datasets, as in the case of UCF-101, which might bias the outcome of this experiment (for instance, in Figure 4.1 we observe that even randomly initialized networks perform better on unseen concepts). These are only a few points that should be taken into consideration in designing novel generalization benchmarks.

For UCF-101, we conclude that all methods perform significantly better on unseen concepts, with the difference in accuracy even surpassing 20% for large amounts of data during training. Moreover, it is noteworthy that *xID* model yields the highest accuracy on unseen concepts, whereas *VICReg-V* shows comparable or even slightly better results on seen concepts provided that the training set is sufficiently large.

In the case of the fully balanced HMDB-51 dataset, we notice once more that performance on unseen concepts is superior compared to their seen counterparts, even though their relative difference is now typically lower than 17%. Furthermore, *xID* clearly outperforms all other models on both seen and unseen concepts; however, *VICReg-V* model is now more competitive on unseen concepts (given enough training examples), since its accuracy is 4% and 10% lower than the top-performing *xID* on unseen and seen concepts, respectively.

In summary, the results depicted in Figures 4.1-4.4 indicate that self-supervised models generalize well on concepts not previously encountered during pre-training; nevertheless, their fit on seen concepts is deemed unsatisfactory. We speculate that this behavior could be attributed to our approach for handling noisy correspondences, since all methods either neglect or aim to mitigate those. However, an optimal solution would leverage such data in a way that would lead to useful gradients (by, e.g. sampling better positives).

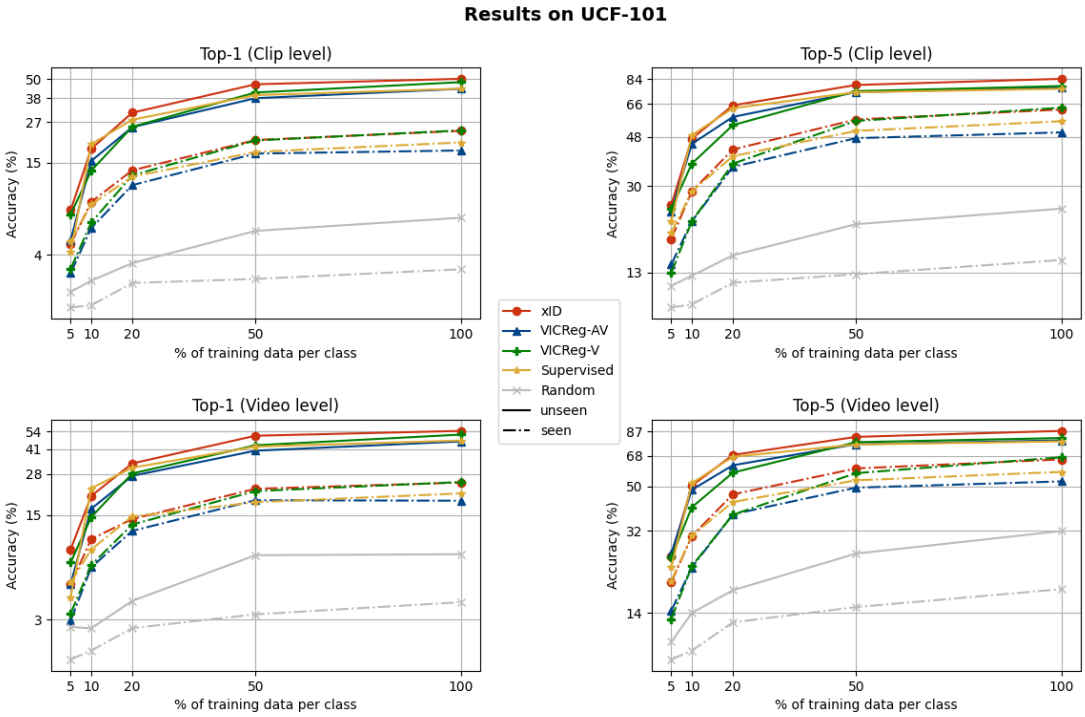

Figure 4.1: Accuracy vs. percentage of training examples per class on UCF-101 (y-axis is in logarithmic scale).

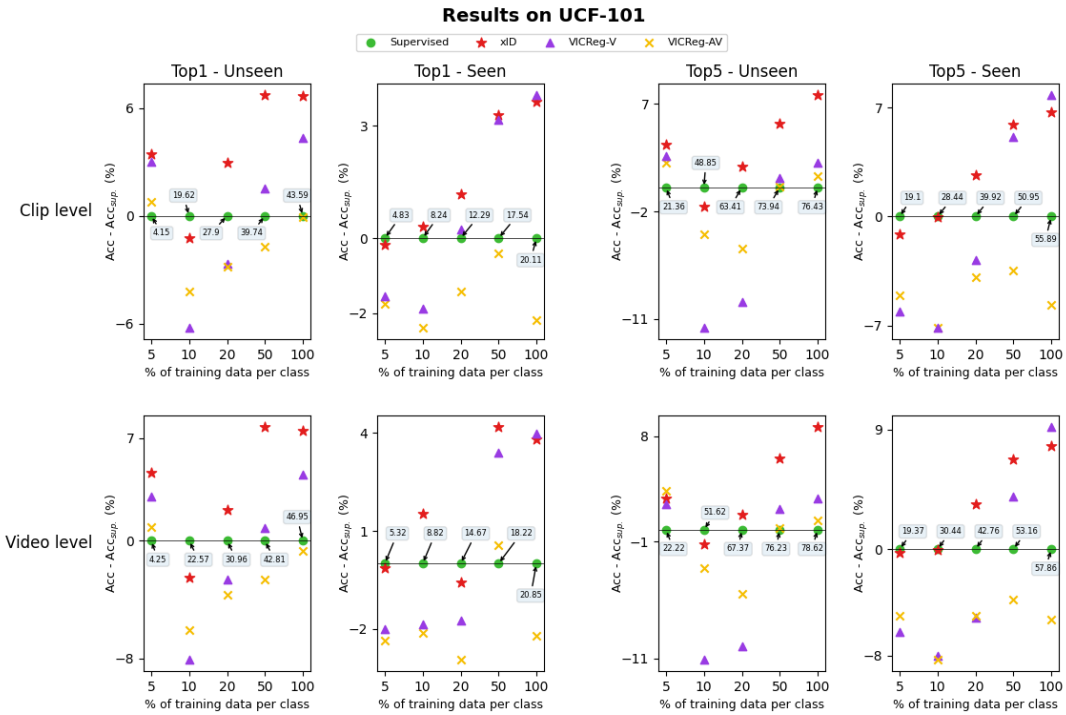

Figure 4.2: Delta accuracy with respect to the supervised baseline for UCF-101. For each percentage, the supervised model's accuracy (in %) is shown inside a box.

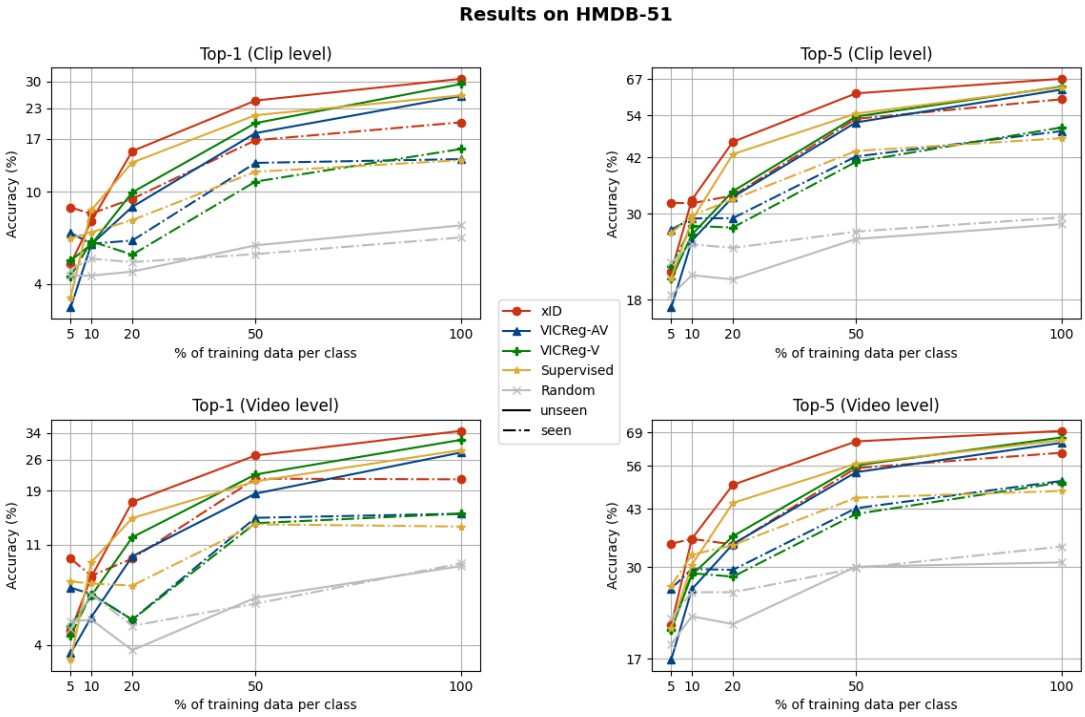

Figure 4.3: Accuracy vs. percentage of training examples per class on HMDB-51 (y-axis is in logarithmic scale).

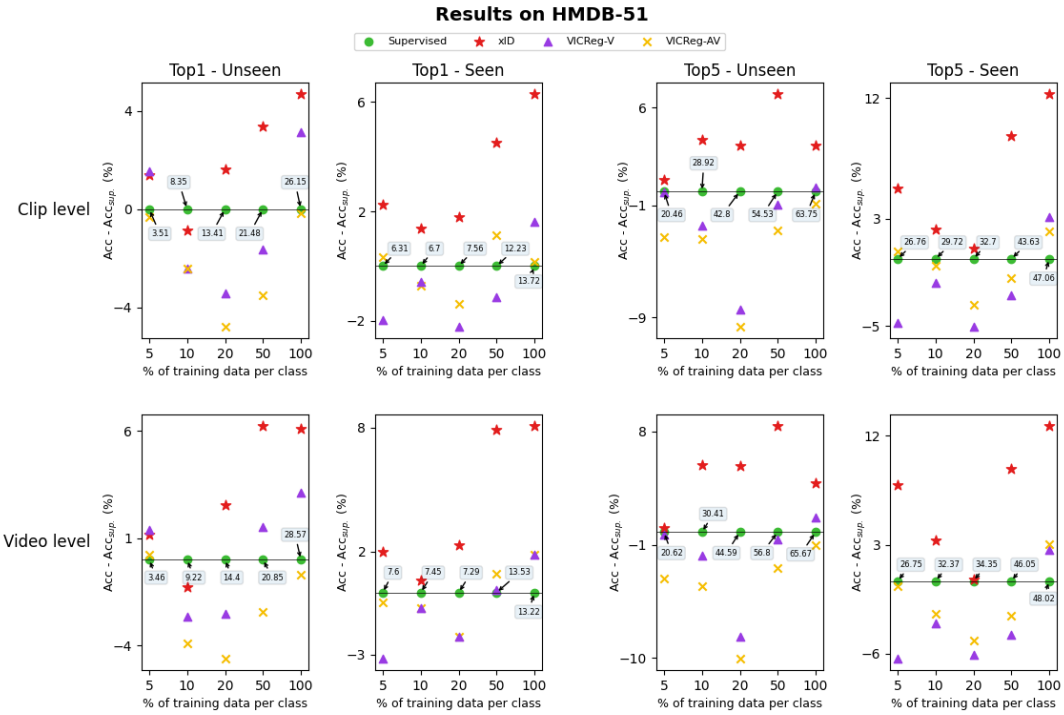

Figure 4.4: Delta accuracy with respect to the supervised baseline for HMDB-51. For each percentage, the supervised model's accuracy (in %) is shown inside a box.

Chapter 5

Conclusion

This study aimed to examine various techniques for learning audio-visual representations in-the-wild via self-supervision. Specifically, the main objective was to systematically compare multi-modal self-supervised approaches with their uni-modal counterparts, as well as a supervised baseline that merges information from both visual and audio streams using late fusion. To this end, we conducted a series of experiments that demonstrated the effectiveness of multi-modal methods, and also provided insight into their weaknesses that should be addressed in future studies.

This analysis was based on two self-supervised learning frameworks, namely cross-modal instance discrimination (*xID*) and variance-invariance-covariance regularization (*VICReg*). The former is an established contrastive learning method that was empirically shown to outperform within-modality discrimination, leading to state-of-the-art results on popular action recognition benchmarks. Moreover, through a modified loss function, *xID* achieves robustness against weakly correspondent inputs, which commonly occur in realistic settings. On the contrary, *VICReg*, which was recently proposed, is the first non-contrastive method (i.e. unlike contrastive approaches, it naturally avoids false negatives during pre-training) that is compatible with multi-modal data; yet, it has not been applied to audio-visual inputs before.

First, using a standard benchmark suite consisting of two different datasets, we experimentally evaluated the extent to which the learned representations are linearly separable and transferable. In this case, *xID* was the top-performing model, followed by a uni-modal *VICReg* model that was only trained on video frames. These results show the importance of eliminating weak audio-visual correspondences, which even deteriorate the supervised baseline's performance. In addition, the multi-modal *VICReg* model gave the worst results, which can be attributed to both the lack of a mechanism that discards noisy inputs and the difference between the learning dynamics of each individual modality. Therefore, introducing a robust regression loss, as well as randomly dropping the audio path to slow down its learning process, is expected to alleviate these issues.

Next, we addressed the problem of generalization, i.e. the models' ability to produce meaningful features when applied to unknown data domains. For this experiment, given the list of labels in the pre-training dataset, each benchmark dataset was split into two disjoint sets, i.e. seen and unseen concepts, while models' performance was assessed on each set of concepts independently. Even in this scenario, *xID* model yielded the best results on both sets of concepts, showing that self-supervised models generalize well to new data. Nevertheless, we also identified some level of underfitting, since accuracy on seen concepts was approximately 20% lower compared to their unseen counterparts. This

effect highlights the importance of extracting strong gradient signals from noisy data as well (removing such data only partially solves the problem).

Last, we tested our best model (*xID*) on an additional task involving video retrieval using all possible combinations of modalities for both queries and candidate videos. Here, we observed that the model yields high retrieval accuracy in the case of a single modality; yet, performance on cross-modal retrieval was no better than random. These results show that the model has not successfully learned to associate objects with their characteristic sounds (also known as source localization).

5.1 Future work

Although self-supervised models show promising results in learning general-purpose representations, it is clear that there is still much room for improvement. One possible direction would involve designing novel approaches for audio-visual self-supervision based on, e.g. multi-task learning. This way, apart from maximizing the agreement between video and audio inputs, the network would also learn to solve more complex tasks such as source localization or separation. In addition, regarding the localization problem mentioned in Appendix A, capsule networks [60], [61] are promising alternatives to standard CNNs in a sense that they encode spatial information better (they discard pooling layers, thus they provide equivariance instead of invariance). In this framework, it is also worth exploring the use of multi-modal capsule network architectures. However, given the techniques currently used in this field, we outline the following avenues for future research.

Unified Architectures The idea of developing modality-agnostic architectures has attracted a lot of attention in the research community, with the recently proposed Perceiver model [62] showing competitive results. However, it is straightforward that early layers should learn modality-specific low-level features (e.g. edges and corners for visual inputs and wavelet-like filters for audio data) which do not necessarily correlate with each other. Concurrently, Nagrani et al. [63] showed the benefits of fusing information between modalities at multiple levels (starting from the middle of each uni-modal network) through a small number of bottlenecks. Thus, it is worth investigating to which extent can we share layers across modalities, as this would result in more efficient architectures, while it would also enable the application of more self-supervised learning methods found in the literature (particularly non-contrastive approaches).

Improved sampling strategies In this thesis, we highlighted the need for extracting strong learning signals even from noisy inputs. To this end, we could leverage generative models (e.g. VQ-VAEs, GANs, diffusion models or any combination thereof) which are capable of producing realistic outputs. Specifically, a viable solution would involve training a conditional generative network that, given an uni-modal input (e.g. video frames), it would output samples from the opposing modality (e.g. audio spectrograms) that are highly similar to this input. In addition, since negative targets yield useful gradients that facilitate learning, we could also form true negative pairs directly in the model's latent space (note that, to achieve this goal, it might be necessary to rearrange the latent space based on an appropriate perceptual metric, e.g. Fréchet inception distance).

Improved evaluation suites Benchmarking audio-visual self-supervised models is crucial for assessing their performance under challenging conditions, i.e. on different datasets and tasks (e.g. recognition, retrieval and localization). However, the most popular evaluation suite in this research area does no longer pose these challenges due to either saturated (UCF-101) or stagnant performance (HMDB-51), while it is not indicative of true gener-

alization performance due to overlap with commonly used pre-training datasets. Hence, inspired by a similar benchmark based on ImageNet [59], we suggest dividing large-scale datasets (e.g. AVA[1] or Kinetics with 400/700 classes[2]) into disjoint sets of semantic categories. This way, for instance, only a single set would be used for pre-training, whereas the rest would serve as downstream datasets. Additionally, this setup would inherently enable measuring generalization, as there would be a clear distinction between seen and unseen concepts. Last, releasing uncurated datasets (e.g. from social media other than YouTube) would also be beneficial for the research community, as it would allow testing self-supervised models in unconstrained real-world applications.

[1] https://research.google.com/ava/
[2] https://www.deepmind.com/open-source/kinetics

Appendix A

Additional results

A.1 Retrieval

Thus far, we focused primarily on the downstream task of action recognition. For the sake of completeness, considering that self-supervised pre-training should yield task-agnostic features, we provide additional results on the task of video retrieval. In this case, the objective is to return a ranked list of candidate videos from the training set which correspond to a specific query, i.e. a video from the test set. Then, the retrieval is considered valid if and only if at least one of the top-k predictions (here, k is either 1 or 5) matches the query's ground-truth label.

For this experiment, we use the subset of UCF-101 that consists of videos with an accompanying audio stream. This subset contains a total of 51 classes, while the length of training and test set is 4797 and 1916, respectively. The choice of this particular subset is motivated by the fact that it enables us to perform both uni-modal and cross-modal retrieval, with the latter being a significantly harder task. Moreover, as a pre-processing step, we extract 0.5 seconds long video clips and 1 second long audio excerpts without augmentation. Note that video (respectively audio) level representations are formed by averaging the feature vectors of 10 uniformly sampled clips per input.

Furthermore, we use our top-performing model here, i.e. the one pre-trained with cross-modal instance discrimination (xID) method. Specifically, all inputs are transformed into representations via their modality specific encoder (either video or audio). Note that both encoders are initialized with their original pre-trained weights and are not further fine-tuned on this downstream task. In the case of training data, we store features, along with their corresponding labels, in GPU memory to efficiently perform the search in the next stage. Once a query (i.e. a sample from the test set) is encountered, the pairwise Euclidean distance between its feature vector and each in-memory representation is computed. After ranking the results in ascending order, only the top-5 predictions are retrieved from the train set and used to calculate the final retrieval accuracy.

In Table A.1 we present the average accuracy per retrieval method. $x \rightarrow y$ notation means the following: using a query (video from test set) with input modality x, we want to retrieve the top-k videos from training set associated with modality y. Additionally, in Figures A.1-A.4 we visualize 10 randomly selected queries along with their corresponding top-5 predictions. For visual clarity, audio spectrograms are replaced with randomly selected frames from the same video. However, in Figures A.5 and A.6, we provide frames along with the corresponding spectrograms. In the following, we conduct a qualitative analysis based on these plots, which provide useful insights on potential failure modes.

Table A.1: Retrieval results (in %) using *xID* model.

Method	Top-1	Top-5
$Video \rightarrow Video$	50.0	68.22
$Audio \rightarrow Audio$	32.15	51.72
$Audio \rightarrow Video$	1.25	4.59
$Video \rightarrow Audio$	3.13	8.77

Based on Table A.1, we notice that uni-modal retrieval, i.e. using the same modality for both queries and candidate videos, yields acceptable results. However, this is not the case for cross-modal retrieval (a task that has not yet been thoroughly explored for audio-visual data), since results are no better than random chance.

Figure A.1 illustrates examples of uni-modal retrieval using visual modality. Clearly, the model produces high-level matches, whereas prediction errors commonly occur in the case of challenging scenes. For instance, from the first two examples of Figure A.1 we infer that although the model successfully identifies the query's semantics (human-object interactions), the retrieved videos contain objects of similar shape and color that alter the content's meaning. Moreover, similar backgrounds could also lead to incorrect matches, as shown in row 6 of Figure A.1. Note that, in these scenarios, audio modality would possibly help disambiguate the predicted actions.

Audio-based retrieval leads to slightly worse results compared to video-only. From Figure A.2 we conclude that the model correctly retrieves musical instruments, while it also adequately discriminates between indoor and outdoor scenes (examples of indoor and outdoor environments can be found in rows 7 and 8 of Figure A.2, respectively). Furthermore, in Figure A.5 we notice that low-quality matches occur either when the query's audio has white noise characteristics (rows 2 and 5) or due to patterns that may not be representative of the action (rows 1 and 3). Last, this figure also shows that the model is relatively insensitive to amplitude (achieved by an appropriate augmentation during pre-training).

On the contrary, *xID* model has subpar performance on cross-modal retrieval. In fact, although there is no clear outcome from Figures A.3, A.4 and A.6, we deduce that audio and video representations are not sufficiently aligned (e.g. in the case of people playing musical instruments, we would expect the model to associate timbre with the instrument's shape or even the position of hands on it). However, it is also worth mentioning that this task is challenging, as UCF-101 contains data involving cluttered scenes and overlapping sound sources (e.g. human speech and background noise).

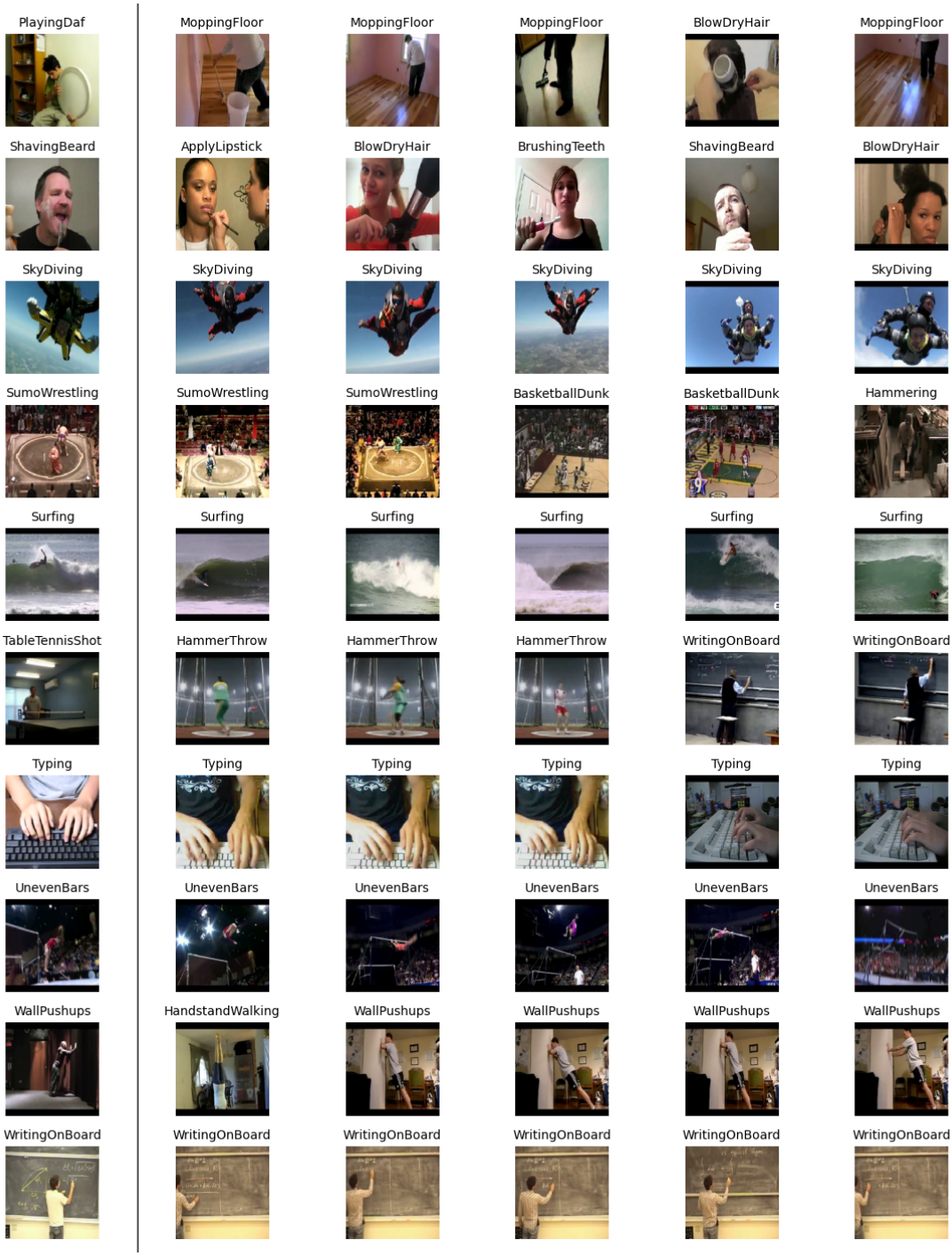

Figure A.1: Randomly selected examples of $Video \rightarrow Video$ retrieval. Each row refers to a different example. The first column (left) depicts the query, while the rest (from left to right) show the model's predictions in descending order of similarity with the query. Ground-truth classes are shown on top of all depicted video frames.

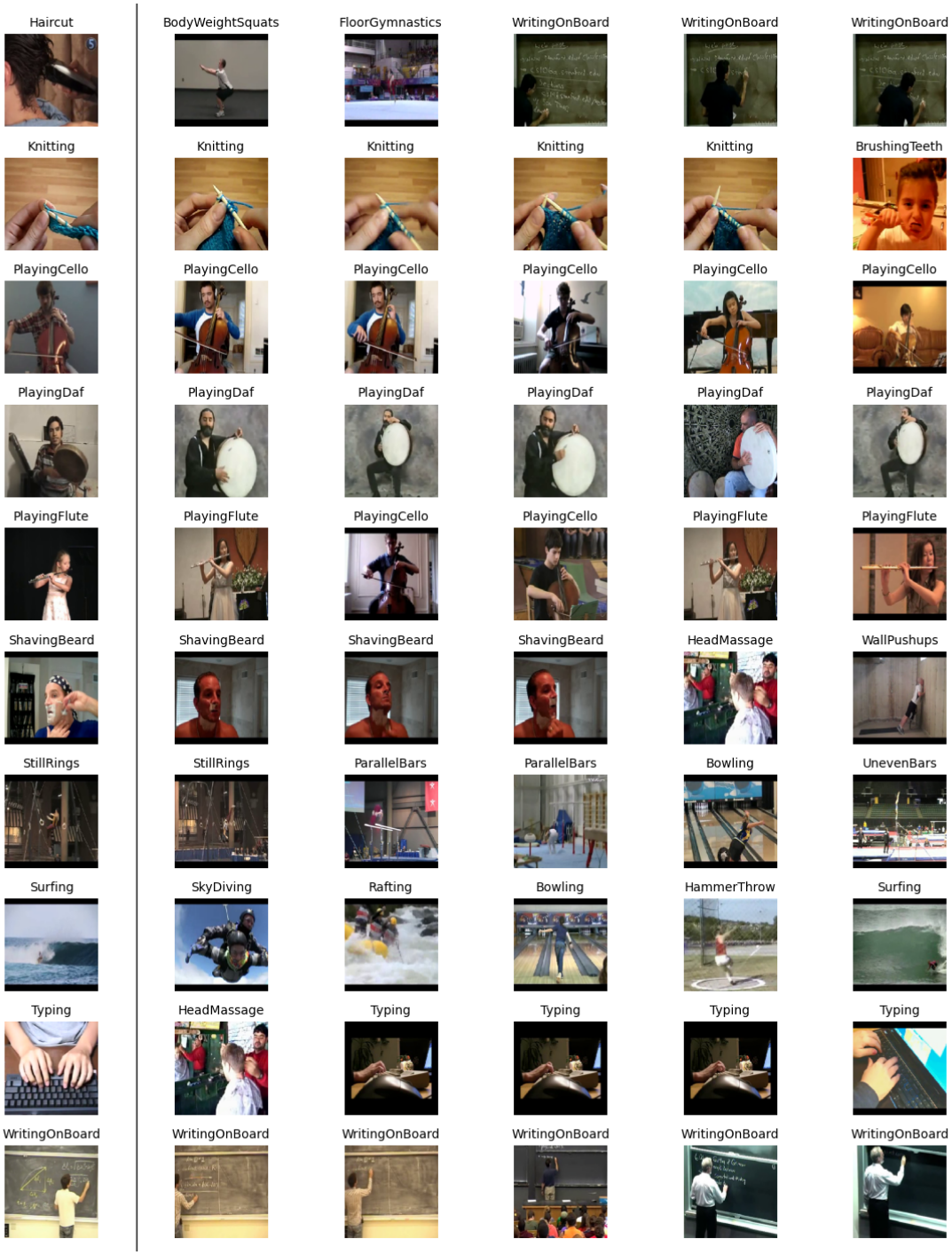

Figure A.2: Randomly selected examples of *Audio* → *Audio* retrieval. Each row refers to a different example. The first column (left) depicts the query, while the rest (from left to right) show the model's predictions in descending order of similarity with the query. Ground-truth classes are shown on top of all depicted video frames.

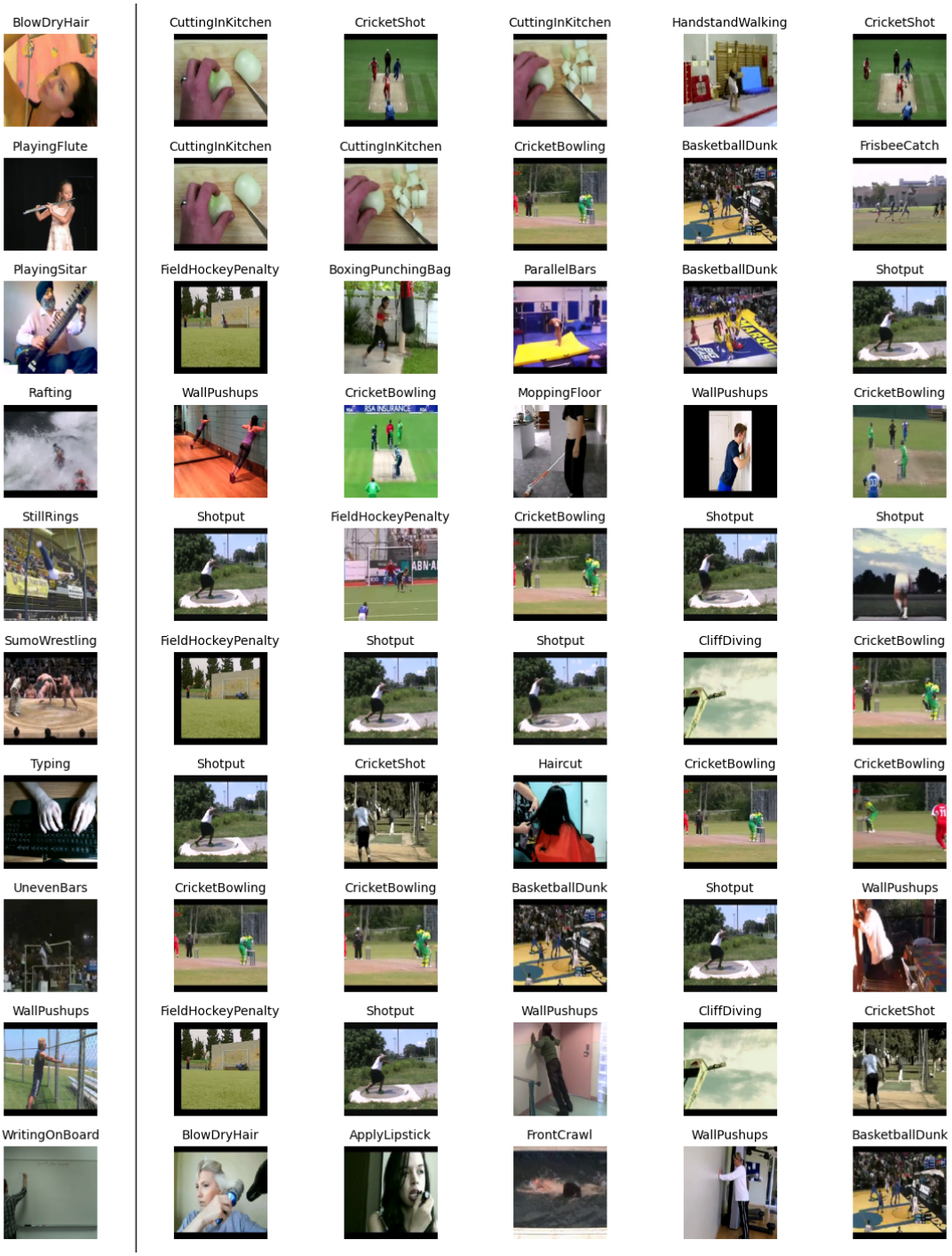

Figure A.3: Randomly selected examples of $Video \rightarrow Audio$ retrieval. Each row refers to a different example. The first column (left) depicts the query, while the rest (from left to right) show the model's predictions in descending order of similarity with the query. Ground-truth classes are shown on top of all depicted video frames.

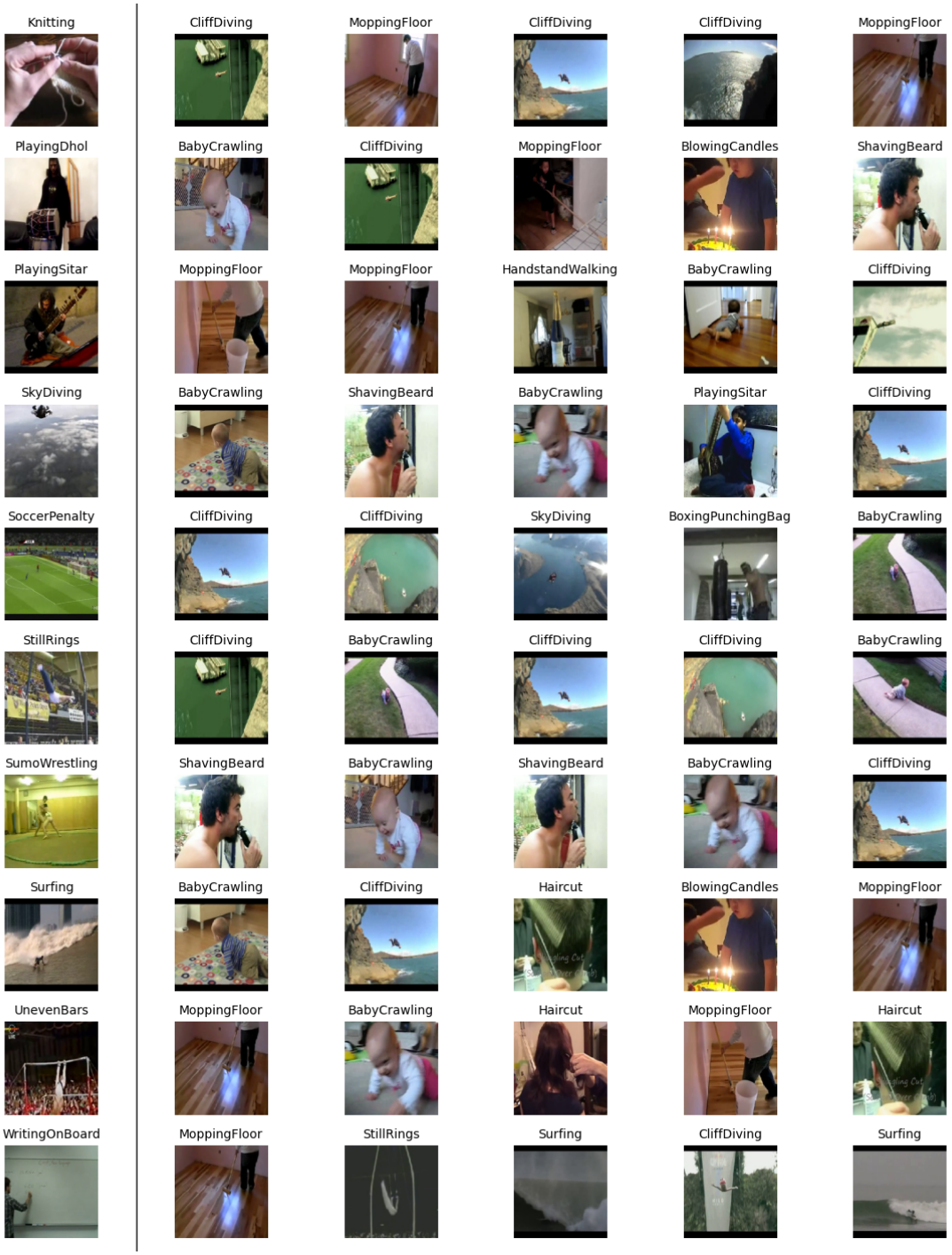

Figure A.4: Randomly selected examples of *Audio* → *Video* retrieval. Each row refers to a different example. The first column (left) depicts the query, while the rest (from left to right) show the model's predictions in descending order of similarity with the query. Ground-truth classes are shown on top of all depicted video frames.

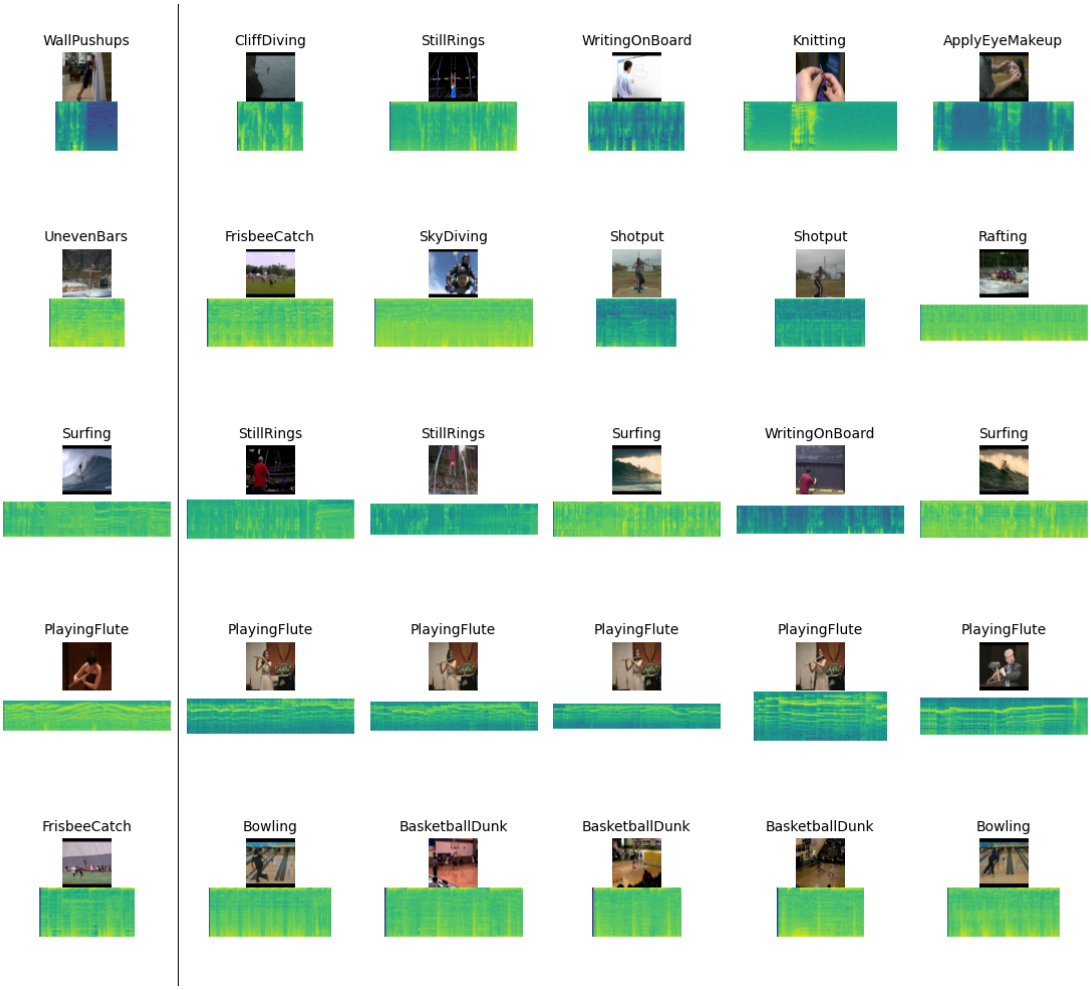

Figure A.5: Additional results of *Audio* → *Audio* retrieval. Each pair of rows refers to a single example. Both queries and predictions are presented with a random frame (upper rows) and their accompanying spectrograms (bottom rows).

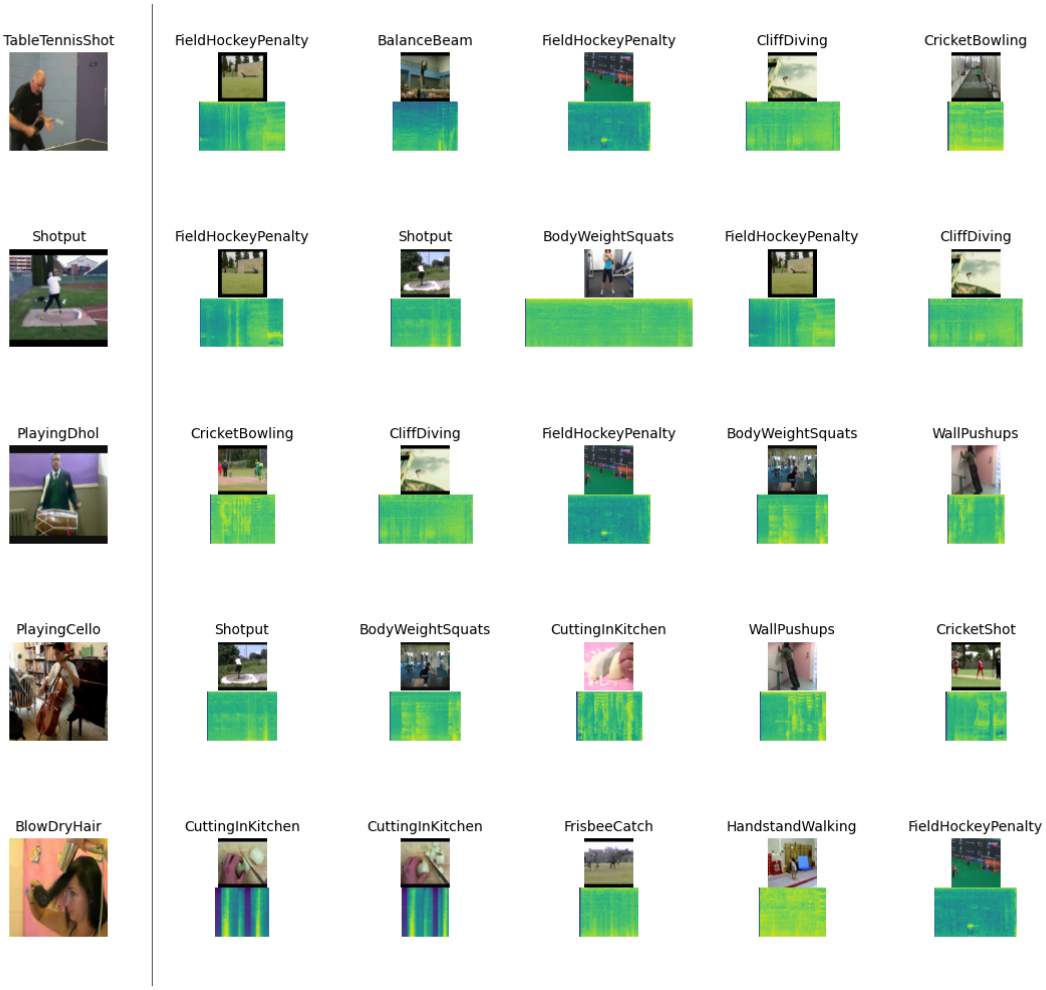

Figure A.6: Additional results of $Video \rightarrow Audio$ retrieval. Each pair of rows refers to a single example. Both queries and predictions are presented with a random frame (upper rows) and their accompanying spectrograms (bottom rows).

Bibliography

[1] D. Kollias, A. Tagaris, A. Stafylopatis, S. Kollias, and G. Tagaris, «Deep neural architectures for prediction in healthcare», *Complex & Intelligent Systems*, vol. 4, no. 2, pp. 119–131, 2018.

[2] A. M. Durrant, G. Leontidis, S. Kollias, *et al.*, «Detection and localisation of multiple in-core perturbations with neutron noise-based self-supervised domain adaptation», 2021.

[3] D. Kollias, P. Tzirakis, M. A. Nicolaou, *et al.*, «Deep affect prediction in-the-wild: Aff-wild database and challenge, deep architectures, and beyond», *International Journal of Computer Vision*, vol. 127, no. 6, pp. 907–929, 2019.

[4] D. Kollias and S. Zafeiriou, «Training deep neural networks with different datasets in-the-wild: The emotion recognition paradigm», in *2018 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2018, pp. 1–8.

[5] ——, «Exploiting multi-cnn features in cnn-rnn based dimensional emotion recognition on the omg in-the-wild dataset», *IEEE Transactions on Affective Computing*, vol. 12, no. 3, pp. 595–606, 2020.

[6] C. Doersch, A. Gupta, and A. A. Efros, «Unsupervised visual representation learning by context prediction», in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1422–1430.

[7] M. Noroozi and P. Favaro, «Unsupervised learning of visual representations by solving jigsaw puzzles», in *European conference on computer vision*, Springer, 2016, pp. 69–84.

[8] R. Zhang, P. Isola, and A. A. Efros, «Colorful image colorization», in *European conference on computer vision*, Springer, 2016, pp. 649–666.

[9] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, «Context encoders: Feature learning by inpainting», in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2536–2544.

[10] R. Zhang, P. Isola, and A. A. Efros, «Split-brain autoencoders: Unsupervised learning by cross-channel prediction», in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1058–1067.

[11] S. Gidaris, P. Singh, and N. Komodakis, «Unsupervised representation learning by predicting image rotations», in *International Conference on Learning Representations*, 2018.

[12] P. Mylonas, E. Spyrou, Y. Avrithis, and S. Kollias, «Using visual context and region semantics for high-level concept detection», *IEEE Transactions on Multimedia*, vol. 11, no. 2, pp. 229–243, 2009.

[13] D. Kollias, M. Yu, A. Tagaris, G. Leontidis, A. Stafylopatis, and S. Kollias, «Adaptation and contextualization of deep neural network models», in *2017 IEEE symposium series on computational intelligence (SSCI)*, IEEE, pp. 1–8.

[14] A. Tagaris, D. Kollias, A. Stafylopatis, G. Tagaris, and S. Kollias, «Machine learning for neurodegenerative disorder diagnosis—survey of practices and launch of benchmark dataset», *International Journal on Artificial Intelligence Tools*, vol. 27, no. 03, p. 1 850 011, 2018.

[15] J. Wingate, I. Kollia, L. Bidaut, and S. Kollias, «Unified deep learning approach for prediction of parkinson's disease», *IET Image Processing*, vol. 14, no. 10, pp. 1980–1989, 2020.

[16] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, «Deep clustering for unsupervised learning of visual features», in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 132–149.

[17] Y. M. Asano, C. Rupprecht, and A. Vedaldi, «Self-labelling via simultaneous clustering and representation learning», *arXiv preprint arXiv:1911.05371*, 2019.

[18] F. De Sousa Ribeiro, F. Calivá, M. Swainson, K. Gudmundsson, G. Leontidis, and S. Kollias, «Deep bayesian self-training», *Neural Computing and Applications*, vol. 32, no. 9, pp. 4275–4291, 2020.

[19] A. Van den Oord, Y. Li, and O. Vinyals, «Representation learning with contrastive predictive coding», *arXiv e-prints*, arXiv–1807, 2018.

[20] O. Henaff, «Data-efficient image recognition with contrastive predictive coding», in *International Conference on Machine Learning*, PMLR, 2020, pp. 4182–4192.

[21] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, «Unsupervised feature learning via non-parametric instance discrimination», in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3733–3742.

[22] Y. Tian, D. Krishnan, and P. Isola, «Contrastive multiview coding», in *European conference on computer vision*, Springer, 2020, pp. 776–794.

[23] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, «Momentum contrast for unsupervised visual representation learning», in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9729–9738.

[24] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, «A simple framework for contrastive learning of visual representations», in *International conference on machine learning*, PMLR, 2020, pp. 1597–1607.

[25] X. Chen, H. Fan, R. Girshick, and K. He, «Improved baselines with momentum contrastive learning», *arXiv preprint arXiv:2003.04297*, 2020.

[26] T. Wang and P. Isola, «Understanding contrastive representation learning through alignment and uniformity on the hypersphere», in *International Conference on Machine Learning*, PMLR, 2020, pp. 9929–9939.

[27] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, «Unsupervised learning of visual features by contrasting cluster assignments», *Advances in Neural Information Processing Systems*, vol. 33, pp. 9912–9924, 2020.

[28] J.-B. Grill, F. Strub, F. Altché, *et al.*, «Bootstrap your own latent-a new approach to self-supervised learning», *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 271–21 284, 2020.

[29] X. Chen and K. He, «Exploring simple siamese representation learning», in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 750–15 758.

[30] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, «Barlow twins: Self-supervised learning via redundancy reduction», in *International Conference on Machine Learning*, PMLR, 2021, pp. 12 310–12 320.

[31] A. Bardes, J. Ponce, and Y. Lecun, «Vicreg: Variance-invariance-covariance regularization for self-supervised learning», in *ICLR 2022-10th International Conference on Learning Representations*, 2022.

[32] R. Arandjelovic and A. Zisserman, «Look, listen and learn», in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 609–617.

[33] ——, «Objects that sound», in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 435–451.

[34] B. Korbar, D. Tran, and L. Torresani, «Cooperative learning of audio and video models from self-supervised synchronization», *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[35] A. Owens and A. A. Efros, «Audio-visual scene analysis with self-supervised multisensory features», in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 631–648.

[36] P. Morgado, Y. Li, and N. Vasconcelos, «Learning representations from audio-visual spatial alignment», *Advances in Neural Information Processing Systems*, vol. 33, pp. 4733–4744, 2020.

[37] H. Alwassel, D. Mahajan, B. Korbar, L. Torresani, B. Ghanem, and D. Tran, «Self-supervised learning by cross-modal audio-video clustering», *Advances in Neural Information Processing Systems*, vol. 33, pp. 9758–9770, 2020.

[38] Y. Asano, M. Patrick, C. Rupprecht, and A. Vedaldi, «Labelling unlabelled videos from scratch with multi-modal self-supervision», *Advances in Neural Information Processing Systems*, vol. 33, pp. 4660–4671, 2020.

[39] P. Morgado, N. Vasconcelos, and I. Misra, «Audio-visual instance discrimination with cross-modal agreement», in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 475–12 486.

[40] P. Morgado, I. Misra, and N. Vasconcelos, «Robust audio-visual instance discrimination», in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 934–12 945.

[41] M. Patrick, P.-Y. Huang, I. Misra, *et al.*, «Space-time crop & attend: Improving cross-modal video representation learning», in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 560–10 572.

[42] M. Patrick, Y. M. Asano, P. Kuznetsova, *et al.*, «On compositions of transformations in contrastive self-supervised learning», in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9577–9587.

[43] K. Soomro, A. R. Zamir, and M. Shah, «Ucf101: A dataset of 101 human actions classes from videos in the wild», *arXiv preprint arXiv:1212.0402*, 2012.

[44] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, «Hmdb: A large video database for human motion recognition», in *2011 International conference on computer vision*, IEEE, 2011, pp. 2556–2563.

[45] Z. Tong, Y. Song, J. Wang, and L. Wang, «Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training», *arXiv preprint*, 2022.

[46] L. Wang, Y. Xiong, Z. Wang, *et al.*, «Temporal segment networks: Towards good practices for deep action recognition», in *European conference on computer vision*, Springer, 2016, pp. 20–36.

[47] L. Sevilla-Lara, Y. Liao, F. Güney, V. Jampani, A. Geiger, and M. J. Black, «On the integration of optical flow and action recognition», in *German conference on pattern recognition*, Springer, 2018, pp. 281–297.

[48] C. Feichtenhofer, H. Fan, B. Xiong, R. Girshick, and K. He, «A large-scale study on unsupervised spatiotemporal representation learning», in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3299–3309.

[49] T. Pan, Y. Song, T. Yang, W. Jiang, and W. Liu, «Videomoco: Contrastive video representation learning with temporally adversarial examples», in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 205–11 214.

[50] I. Dave, R. Gupta, M. N. Rizve, and M. Shah, «Tclr: Temporal contrastive learning for video representation», *Computer Vision and Image Understanding*, vol. 219, p. 103 406, 2022.

[51] R. Qian, T. Meng, B. Gong, *et al.*, «Spatiotemporal contrastive video representation learning», in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6964–6974.

[52] S. N. Gowda, M. Rohrbach, and L. Sevilla-Lara, «Smart frame selection for action recognition», *arXiv preprint arXiv:2012.10671*, 2020.

[53] C.-Y. Chuang, R. D. Hjelm, X. Wang, *et al.*, «Robust contrastive learning against noisy views», *arXiv preprint arXiv:2201.04309*, 2022.

[54] H. Chen, W. Xie, A. Vedaldi, and A. Zisserman, «Vggsound: A large-scale audio-visual dataset», in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2020, pp. 721–725.

[55] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, «A closer look at spatiotemporal convolutions for action recognition», in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 6450–6459.

[56] A. Paszke, S. Gross, F. Massa, *et al.*, «Pytorch: An imperative style, high-performance deep learning library», *Advances in neural information processing systems*, vol. 32, 2019.

[57] D. P. Kingma and J. Ba, «Adam: A method for stochastic optimization», *arXiv preprint arXiv:1412.6980*, 2014.

[58] F. Xiao, Y. J. Lee, K. Grauman, J. Malik, and C. Feichtenhofer, «Audiovisual slowfast networks for video recognition», *arXiv preprint arXiv:2001.08740*, 2020.

[59] M. B. Sariyildiz, Y. Kalantidis, D. Larlus, and K. Alahari, «Concept generalization in visual representation learning», in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9629–9639.

[60] F. D. S. Ribeiro, G. Leontidis, and S. Kollias, «Capsule routing via variational bayes», in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 3749–3756.

[61] F. De Sousa Ribeiro, G. Leontidis, and S. Kollias, «Introducing routing uncertainty in capsule networks», *Advances in Neural Information Processing Systems*, vol. 33, pp. 6490–6502, 2020.

[62] A. Jaegle, F. Gimeno, A. Brock, O. Vinyals, A. Zisserman, and J. Carreira, «Perceiver: General perception with iterative attention», in *International conference on machine learning*, PMLR, 2021, pp. 4651–4664.

[63] A. Nagrani, S. Yang, A. Arnab, A. Jansen, C. Schmid, and C. Sun, «Attention bottlenecks for multimodal fusion», *Advances in Neural Information Processing Systems*, vol. 34, pp. 14 200–14 213, 2021.