NATIONAL TECHNICAL UNIVERSITY OF ATHENS

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

DIVISION OF SIGNALS, CONTROL AND ROBOTICS

Emergent Object-Centric Perception Through Intrinsically Motivated Play

DIPLOMA THESIS

of

Orestis Konstantaropoulos

Supervisor: Petros Maragos
Professor, NTUA

Computer Vision, Speech Communication & Signal Processing Group
Athens, June 2025

National Technical University of Athens
School of Electrical and Computer Engineering
Division of Signals, Control and Robotics
Computer Vision, Speech Communication and Signal Processing Group

Emergent Object-Centric Perception Through Intrinsically Motivated Play

DIPLOMA THESIS

OF

ORESTIS
KONSTANTAROPOULOS

Supervisor: Petros Maragos
Professor, NTUA
Co-Supervisor: Giorgos Retsinas
Researcher, Athena Research Center

Approved by the examination committee on 18th June, 2025.

.....	.....	.....
Petros Maragos	Gerasimos Potamianos	Ioannis Kordonis
Professor NTUA	Assoc Professor UTH	Asst Professor NTUA

Athens, June 2025.

.....
ORESTIS KONSTANTAROPOULOS
Graduate of Electrical and
Computer Engineering NTUA

Περίληψη

Σε αντίθεση με τα τυπικά συστήματα όρασης υπολογιστών που βασίζονται στην παθητική επεξεργασία δεδομένων, στη βιολογία η μάθηση συμβαίνει συχνότερα μέσω αλληλεπίδρασης. Τα παιδιά, για παράδειγμα, αλληλεπιδρούν για ώρες με τα παιχνίδια τους και εξερευνούν το περιβάλλον τους, χωρίς συγκεκριμένο στόχο, με φαινομενικά τυχαίο τρόπο. Παράλληλα, φτιάχνουν μοντέλα μετάβασης, με τα οποία μπορούν να προβλέπουν τις αλλαγές που επιφέρουν οι πράξεις τους στο περιβάλλον τους. Τα μοντέλα αυτά τα χρησιμοποιούν αργότερα για να μάθουν γρηγορότερα νέες δεξιότητες. Σε αυτή τη διαδικασία τις περισσότερες φορές απουσιάζει άμεση επίβλεψη που κατευθύνει τη μάθηση. Αντίθετα, η μάθηση στηρίζεται περισσότερο στα εσωτερικά κίνητρα των οργανισμών και στα δομικά χαρακτηριστικά των αισθητήριων οργάνων τους.

Έτσι προκύπτει ένα θεμελιώδες ερευνητικό ερώτημα: Μπορεί, παρόμοια, ένα ρομπότ να αναπτύξει μια κατανόηση του περιβάλλοντος του αξιοποιώντας μόνο την ικανότητα του να αλληλεπιδρά με αυτό, χωρίς εκ των προτέρων γνώση, ή εξωτερική επίβλεψη. Στη παρούσα διπλωματική εργασία εξετάζουμε πώς τεχνητοί δράστες μπορούν να εξερευνήσουν και να μάθουν το περιβάλλον τους αυτόνομα, βασιζόμενοι σε εσωτερικά κίνητρα.

Προτείνουμε, λοιπόν, μια νέα, πλήρως αυτο-επιβλεπόμενη και αντικειμενο-κεντρική προσέγγιση. Το σύστημα μας πρώτα διακρίνει το χώρο του σε διακριτές οντότητες-αντικείμενα χρησιμοποιώντας αυτο-επιβλεπόμενους και αντικειμενο-κεντρικούς αλγορίθμους όρασης υπολογιστών πάνω σε δεδομένα που έχουν συλλεχθεί από τυχαίες δράσεις ενός ρομποτικού βραχίονα. Στη συνέχεια, ένα μοντέλο μετάβασης βασισμένο σε γράφους εκπαιδεύεται να προβλέπει τις μελλοντικές καταστάσεις των οντοτήτων αυτών. Ωστόσο, λόγω της περιορισμένης ποικιλίας των δεδομένων που βασίζονται σε τυχαίες δράσεις, το μοντέλο μετάβασης αδυνατεί να προβλέψει σωστά την κίνηση των αντικειμένων.

Για αυτό, σχεδιάζουμε ένα σήμα επιβράβευσης που βασίζεται στο σφάλμα πρόβλεψης του μοντέλου μετάβασης. Πάνω σε αυτό το σήμα εκπαιδεύουμε μια πολιτική η οποία τελικά προτείνει πιο ενδιαφέρουσες δράσεις στο βραχίονα, δράσεις που προκαλούν τρεις φορές περισσότερη κίνηση των αντικειμένων σε σύγκριση με τις τυχαίες. Τέλος, εκπαιδεύουμε περαιτέρω τα μοντέλα όρασης και μετάβασης χρησιμοποιώντας νέα δεδομένα που συλλέγουμε με την νέα πολιτική. Τα μοντέλα τώρα παρουσιάζουν βελτίωση τόσο στην ικανότητα αναπαράστασης και ανακατασκευής του χώρου όσο και στην ικανότητα πρόβλεψης τις κίνησης των αντικειμένων.

Επαληθεύουμε την μέθοδο μας σε ένα περιβάλλον προσομοίωσης και δείχνουμε ότι μέσω της αυτο-επιβλεπόμενης αλληλεπίδρασης μπορούν τελικά να προκύψουν χρήσιμες οπτικές αναπαραστάσεις. Η παρούσα διπλωματική αποτελεί ένα ακόμα παράδειγμα του πως η μελέτη της νόησης όπως συναντάται στη βιολογία, αλλά και της αναπτυξιακής ψυχολογίας μπορούν να συνεισφέρουν στη σχεδίαση και τη βελτίωση των συστημάτων τεχνητής νοημοσύνης. Συγκεκριμένα, δείχνουμε ότι η αντικειμενο-κεντρική μάθηση, βασισμένη σε εσωτερικά κίνητρα μπορεί να συνεισφέρει στην αυτόνομη ανάπτυξη συστημάτων κατανόησης του κόσμου. Τα συστήματα αυτά θα είναι σε θέση να αναπτύσσονται διαρκώς και αυτόνομα σε νέα περιβάλλοντα.

Τμήμα της εργασίας έγινε δεκτό στο συνέδριο της IEEE, International Conference on Development and Learning (ICDL) Prague, 2025 με τίτλο "Push, See, Predict: Emergent Perception Through Intrinsically Motivated Play" [1] και συγγραφείς τους Orestis Konstantaropoulos, Mehdi Khamassi, Petros Maragos και George Retsinas.

Λεξεις κλειδιά - Αντικειμενο-κεντρική Όραση Υπολογιστών, Εσωτερικά Παρακινούμενη Ενισχυτική Μάθηση, Active Perception, Μοντέλα Μετάβασης, Βαθιά Μάθηση, CNNs, GNNs, ViTs

Abstract

Unlike conventional vision systems that rely on passive observation, biological agents can learn through physical interaction. Human infants, for example, spend hours interacting with toys in seemingly random ways—exploring their environment and engaging in non-goal-directed behaviors. It is believed that such agents construct internal transition models that allow them to predict the future states of their environment, which they later use to efficiently acquire new skills. This process typically unfolds in the absence of explicit supervision. Instead, biological learning is driven by intrinsic incentives and shaped by structural inductive biases that help the agent make sense of its surroundings.

This raises a fundamental question: Can a robot similarly develop an understanding of its environment purely through interaction, without any prior knowledge or external supervision? In this thesis, we investigate how artificial agents can autonomously explore and learn about their environment through intrinsic motivation, much like how children engage in curious free play.

To this end, we propose a novel, fully self-supervised, object-centric learning framework. Our system first segments visual input into discrete entities using Slot Attention, a self-supervised object-centric vision model trained entirely on data collected from random actions of a robotic arm. A graph-based world model is then trained to predict object-centric dynamics. However, due to the limited diversity of interactions in the initial dataset, the model struggles to capture object motion.

To overcome this, we introduce an intrinsically motivated reward signal based on world model's prediction error. This reward guides a policy that actively collects informative trajectories by proposing actions that are more likely to challenge the current model's predictions. Empirically, this policy proposes actions that result in up to three times more object displacement compared to random actions, leading to significantly richer training data. We then fine-tune both the vision and world model on these data, which leads to improved prediction and reconstruction performance. We validate our method in a simulated robotic environment with diverse objects, demonstrating that meaningful visual and physical representations can emerge entirely from self-supervised interaction.

The findings of this thesis contribute to the growing body of cognitively inspired algorithms designed to enhance artificial learning systems. Specifically, this thesis highlights the potential of intrinsically motivated, object-centric learning for autonomous world perception and modeling; paving the way for the designing of systems that can incrementally develop in novel, open-ended environments without human supervision.

Part of our work was accepted at the 2025 IEEE International Conference on Development and Learning (ICDL) Prague, titled "Push, See, Predict: Emergent Perception Through Intrinsically Motivated Play" [1] with the authors being Orestis Konstantaropoulos, Mehdi Khamassi, Petros Maragos and George Retsinas.

Keywords - Object-Centric Computer Vision, Intrinsically Motivated Reinforcement Learning, Active Perception, World Models, Deep Learning, CNNs, GNNs, ViTs

Ευχαριστίες

Η ολοκλήρωση της παρούσας εργασίας σηματοδοτεί το τέλος των προπτυχιακών σπουδών μου στο Εθνικό Μετσόβιο Πολυτεχνείο. Στο πλαίσιο των σπουδών αυτών, αλλά και της παρούσας εργασίας, ήταν απαραίτητη η επαφή, η εξοικείωση και η εμβάθυνση σε πολλές και σύνθετες έννοιες από τα διάφορα πεδία των επιστημών του Ηλεκτρολόγου Μηχανικού, του Μηχανικού Υπολογιστών και όχι μόνο. Η διαδικασία αυτή δεν θα μπορούσε να στεφθεί με επιτυχία χώρις την συνεισφορά των καθηγητών μου, των συμφοιτητών μου αλλά και της οικογένειας μου.

Συγκεκριμένα, θα ήθελα να ευχαριστήσω τον καθηγητή Πέτρο Μαραγκό για την ευκαιρία να εκπονήσω την διπλωματική μου εργασία στο εργαστήριο Όρασης Υπολογιστών και Επεξεργασίας Σήματος σε ένα διεπιστημονικό θέμα. Επίσης, ευχαριστώ θερμά τον Δρ. Γιώργο Ρετσινά για την καθοριστική συμβολή του σε όλη τη διάρκεια της εργασίας. Από τις ουσιαστικές συζητήσεις και τη διαμόρφωση των αρχικών ιδεών, έως την πολύτιμη υποστήριξή του σε καθημερινές τεχνικές δυσκολίες, η καθοδήγησή του υπήρξε καθοριστική. Ακόμα, ευχαριστώ τον Mehdi Khamassi, ερευνητή από το Πανεπιστήμιο της Σορβόννης, για την συνεισφορά του στη συγγραφή του άρθρου που γράφτηκε με αφορμή την διπλωματική.

Τέλος, θα ήθελα να ευχαριστήσω τους φίλους μου, εντός και εκτός σχολής, καθώς είναι εκείνοι που δίνουν νόημα και αξία σε κάθε προσπάθεια μου, αλλά και τους γονείς μου Γιάννη και Ασημίνα και την αδερφή μου Λυδία για την άνευ όρων και αδιάλειπτη στήριξη τους.

Ορέστης Κωνσταναρόπουλος
Ιούνιος 2025

Contents

List of Figures

List of Tables

Εκτεταμένη Ελληνική Περίληψη

Εισαγωγή

Τα σημερινά μοντέλα τεχνητής νοημοσύνης (ΤΝ) επιτυγχάνουν ολοένα και καλύτερες επιδόσεις σε προβλήματα αναγνώρισης εικόνας και βίντεο, αλλά και σε προβλήματα απόφασης. Παρά την εντυπωσιακή πρόοδο, τα σύγχρονα μοντέλα μηχανικής μάθησης (ΜΜ) ακόμα δεν φτάνουν τις νοητικές και αντιληπτικές ικανότητες των βιολογικών οργανισμών. Οι άνθρωποι χρειάζονται πολύ μικρή εξάσκηση και λίγα δεδομένα για να καταφέρουν να μάθουν να προσαρμοστούν σε ένα νέο περιβάλλον και να εκτελέσουν μια νέα εργασία. Αντίθετα τα συστήματα ΜΜ εξαρτώνται από τη διαθεσιμότητα μεγάλου αριθμού δεδομένων και σημάτων επίβλεψης σχεδιασμένα από ειδικούς ανθρώπους. Τα δεδομένα αυτά πρέπει να είναι ανεξάρτητα και ομοιόμορφα κατανεμημένα καθώς συχνά τα συστήματα ΜΜ αδυνατούν να γενικεύσουν αποτελεσματικά όταν η κατανομή των δεδομένων αλλάζει. Για παράδειγμα οι άνθρωποι μπορούν εύκολα να πλοηγηθούν σε μέρη που δεν έχουν βρεθεί στο παρελθόν, να χειριστούν αντικείμενα που δεν έχουν ξαναδεί και να αποκτήσουν νέες δεξιότητες γρήγορα εκμεταλλευόμενοι τη γνώση που έχουν εξάγει από προηγούμενες εμπειρίες τους [22, 23]. Για αυτό, η μελέτη της βιολογικής νοημοσύνης και των αρχών της κρίνεται απαραίτητη στην προσπάθεια μας να σχεδιάσουμε πιο αποτελεσματικά και εύρωστα τεχνητά συστήματα.

Στόχος της διπλωματικής

Στην παρούσα διπλωματική προσεγγίζουμε την ενεργητική αντίληψη, active perception, εμπνεόμενοι από αρχές των γνωστικών διεργασιών και εκμεταλλευόμενοι τις τελευταίες εξελίξεις στο πεδίο της Μηχανικής Μάθησης. Ο στόχος μας είναι να μιμηθούμε τη συμπεριφορά ενός μικρού παιδιού που αντιλαμβάνεται τον κόσμο του, αλληλεπιδρά με αυτόν, διαδοχικά βελτιώνει την αντίληψη του και αναπτύσσει ένα εσωτερικό μοντέλο που τον εξηγεί. Σε αντίθεση με τα περισσότερα συστήματα ΤΝ που βασίζονται σε σήματα επίβλεψης και μεγάλα σύνολα δεδομένων, στοχεύουμε να αναπτύξουμε μοντέλα που μπορούν να εκπαιδευτούν μόνα τους σε καινούργια,

ανοιχτά περιβάλλοντα. Για να το πετύχουμε αυτό, προτείνουμε μια διαδικασία εκπαίδευσης που στηρίζεται σε εδραιωμένες αρχές από την αναπτυξιακή ψυχολογία, την γνωσιακή επιστήμη και την τεχνητή νοημοσύνη.

Στόχος μας είναι να αναπτύξουμε από το μηδέν ένα σύστημα με ικανότητες οπτικής αντίληψης. Για αυτό, θα χρησιμοποιήσουμε την ικανότητα του συστήματος μας να αλληλεπιδρά με το περιβάλλον του. Θα υιοθετήσουμε μια αντικειμενο-κεντρική προσέγγιση, ενώ θα κατασκευάσουμε μοντέλα μετάβασης που ενσωματώνουν τα δυναμικά χαρακτηριστικά του κόσμου του συστήματος. Το σύστημα αυτό θα αναπτύσσεται αυξητικά. Οδηγούμενο από εσωτερική περιέργεια, με την πάροδο του χρόνου θα βελτιώνει τα μοντέλα όρασης και μετάβασης του.

Σημαντικές αρχές από την αναπτυξιακή ψυχολογία
Η κύρια αρχή στην οποία στηρίζουμε τη προσέγγιση μας είναι αυτή της **ενσώματης νόησης**, η οποία υποστηρίζει ότι η νοημοσύνη προκύπτει μέσω της αλληλεπίδρασης του δράστη με το περιβάλλον του. Είναι δηλαδή αποτέλεσμα της δραστηριότητας που συνδυάζει την αισθητηριακή λήψη πληροφοριών και την κινητική απόκριση [24]. Τα βρέφη για παράδειγμα δεν γεννιούνται με προηγμένες νοητικές ικανότητες, αλλά τις αναπτύσσουν εξερευνώντας και αλληλεπιδρώντας με τον κόσμο τους. Μεγαλώνουν μέσα σε έναν κόσμο γεμάτο επαναλαμβανόμενα πρότυπα και κανονικότητες, οι οποίες διαμορφώνουν σταδιακά την αντίληψη, τις πράξεις και τη σκέψη τους. Η νοημοσύνη τους δεν είναι κάτι απομονωμένο, αλλά πηγάζει και εξελίσσεται μέσα από τις εμπειρίες τους και τη συνεχή αλληλεπίδραση με το περιβάλλον. Με βάση αυτή τη θεώρηση, επιδιώκουμε να καλλιεργήσουμε αντίστοιχες ικανότητες αντίληψης και μάθησης σε ένα τεχνητό σύστημα, δίνοντάς του τη δυνατότητα να δράσει και να πειραματιστεί μέσα στο δικό του περιβάλλον. Όπως και στα βρέφη, έτσι και σε ένα τέτοιο σύστημα, η ελεύθερη εξερεύνηση, ακόμη κι αν φαίνεται τυχαία ή χωρίς συγκεκριμένο σκοπό, μπορεί να ενισχύσει τη μάθηση και την ανάπτυξη νοημοσύνης.

Θα χρησιμοποιήσουμε ακόμα την έννοια των **μοντέλων μετάβασης** [25, 26]. Οι άνθρωποι διαμορφώνουν εσωτερικές αναπαραστάσεις του κόσμου με βάση αισθητηριακές εμπειρίες, οι οποίες καθοδηγούν την αντίληψη, τη δράση και τη λήψη αποφάσεων. Τέτοια μοντέλα επιτρέπουν στα ζώα, δυνητικά και στα τεχνητά συστήματα, να προβλέπουν αποτελέσματα ενεργειών, να σχεδιάζουν, να εξερευνούν και να επιλύουν προβλήματα. Στα πλαίσια αυτής της διπλωματικής, εκπαιδεύουμε μοντέλα μετάβασης με μη εποπτευόμενο τρόπο, ώστε να ενισχύσουμε την εξερεύνηση και να βελτιώσουμε σταδιακά τόσο την αντίληψη όσο και τα ίδια τα μοντέλα [27].

Παράλληλα, υιοθετούμε μια **αντικειμενο-κεντρική προσέγγιση**. Η ικανότητα οργάνωσης του οπτικού ερεθίσματος σε διακριτά αντικείμενα αποτελεί θεμελιώδες χαρακτηριστικό της ανθρώπινης αντίληψης. Από τη βρεφική ηλικία, οι άνθρωποι αναγνωρίζουν αντικείμενα που κινούνται με συνεκτικό τρόπο και μαθαίνουν τις ιδιότητές τους μέσα από την αλληλεπίδραση. Αντί να αντιμετωπίζουμε αυτήν την διάκριση του οπτικού ερεθίσματος σε αντικείμενα ως μια παθητική διαδικασία, επιδιώκουμε ένα σύστημα που μαθαίνει μέσω δράσης: ξεκινάμε από την κατάτμηση του οπτικού πεδίου σε διακριτές οντότητες και προχωράμε προς την κατανόηση των φυσικών χαρακτηριστικών των οντοτήτων και την εξαγωγή αναπαραστάσεων, με τελικό στόχο

τη χρήση αυτών των αναπαραστάσεων σε προβλήματα ελέγχου.

Περίληψη της προσέγγισης μας
Ο στόχος μας είναι να ερευνήσουμε εάν ένα μοντέλο μπορεί να αναπτύξει αυξητικά αντίληψη του κόσμου του αμιγώς μέσω της αλληλεπίδρασης με αυτόν, χωρίς να βασίζεται σε εξωτερική επίβλεψη ή μεγάλα σύνολα δεδομένων. Για να απαντήσουμε σε αυτό το ερώτημα σχεδιάζουμε ένα περιβάλλον στο οποίο ένας ρομποτικός βραχίονας πάνω σε ένα τραπέζι εξερευνά ενεργητικά αντικείμενα που βρίσκονται μπροστά του με στόχο να μεγιστοποιήσει ένα σήμα επιβράβευσης που παράγει το ίδιο το ρομπότ. Οι περισσότερες σχετικές εργασίες έως τώρα αξιοποιούν ένα στάσιμο προκαθορισμένο μοντέλο για την επεξεργασία του οπτικού ερεθίσματος [28, 29]. Αντίθετα εμείς χρησιμοποιούμε τα δεδομένα που συλλέγονται κατά τη διάρκεια της αλληλεπίδρασης για να βελτιώσουμε τα μοντέλα όρασης και μετάβασης. Τελικά, καταφέρνουμε να εκπαιδεύσουμε τα μοντέλα αυτά από το μηδέν. Σχεδιάζουμε ένα εσωτερικά παρακινούμενο σήμα επιβράβευσης που οδηγεί το σύστημα να επιλέγει δράσεις που προκαλούν έως και τρεις φορές περισσότερη κίνηση των αντικείμενων του τραπεζιού σε σύγκριση με τυχαίες δράσεις. Δείχνουμε, ακόμα, ότι η πολίτικη που ακολουθεί το σύστημα οδηγεί στη βελτίωση της ικανότητας πρόβλεψης του μοντέλου μετάβασης αλλά και της ανακατασκευής εικόνας από το μοντέλο όρασης.

Συνεισφορά στη κοινότητα
Για να επαληθεύσουμε τις προτεινόμενες μεθόδους, διεξάγουμε πειράματα με ένα ρομποτικό βραχίονα σε ένα περιβάλλον προσομοίωσης. Τα αποτελέσματα μας δείχνουν ότι η μέθοδος μας κρίνεται αποτελεσματική σε σενάρια όπου δεν υπάρχει σήμα επίβλεψης, προεκπαιδευμένα μοντέλα ή εξωτερικά σύνολα δεδομένων.

Συνοπτικά, οι βασικές μας συνεισφορές είναι οι εξής:

1. **Αντικειμενο-κεντρική Μοντελοποίηση του Κόσμου:** Υιοθετούμε μια αντικειμενο-κεντρική προσέγγιση και αναπτύσσουμε ένα μοντέλο μετάβασης που προβλέπει την μελλοντική κατάσταση των αναπαραστάσεων των αντικειμένων. Σε αντίθεση με τις περισσότερες μεθόδους στη βιβλιογραφία που λειτουργούν στον υψηλής διαστατικότητας οπτικό χώρο, το δικό μας μοντέλο κάνει προβλέψεις σε έναν μικρότερο, λανθάνοντα χώρο, επιτρέποντας αποδοτικότερες και πιο δομημένες προβλέψεις.

2. **Πλήρως Αυτο-Εποπτευόμενη Μάθηση από το Μηδέν:** Εκπαιδεύουμε τόσο το μοντέλο όρασης όσο και το μοντέλο μετάβασης πλήρως από το μηδέν, χωρίς εποπτεία ή εξωτερικά σύνολα δεδομένων. Χρησιμοποιώντας προηγμένες τεχνικές αυτο-εποπτευόμενης μάθησης, τα μοντέλα εκπαιδεύονται αποκλειστικά με δεδομένα που συλλέγει αυτόνομα το ρομπότ μέσω αλληλεπίδρασης.

3. **Εσωτερική Παρακίνηση του Συστήματος μέσω της Αβεβαιότητας Πρόβλεψης:** Σχεδιάζουμε ένα καινοτόμο και εσωτερικά παρακινούμενο σήμα επιβράβευσης, βασισμένο στο σφάλμα πρόβλεψης του μοντέλου μετάβασης. Το σήμα αυτό φιλτράρει το θόρυβο που εισάγουν τα μοντέλα και παρακινεί πολιτικές που οδηγούν σε έως και τριπλάσια με-

τακίνηση των αντικειμένων.

4. **Ενεργητική Εξερεύνηση για τη Σταδιακή Βελτίωση των Μοντέλων:** Δείχνουμε ότι η περαιτέρω εκπαίδευση των μοντέλων σε δεδομένα που συλλέγονται από την εκπαιδευμένη πολιτική οδηγεί σε σημαντική βελτίωση της κατανόησης του κόσμου από το ρομπότ. Η βελτίωση αυτή αποτυπώνεται ποσοτικά μέσω της επίδοσης σε πρόβλεψη και οπτική ανακατασκευή, τόσο για το μοντέλο όρασης όσο και για το μοντέλο μετάβασης.

5. **Εκτενής Πειραματική Αξιολόγηση:** Αξιολογούμε την προσέγγισή μας σε ρομποτικό περιβάλλον προσομοίωσης, επιδεικνύοντας ουσιαστικές βελτιώσεις των μοντέλων

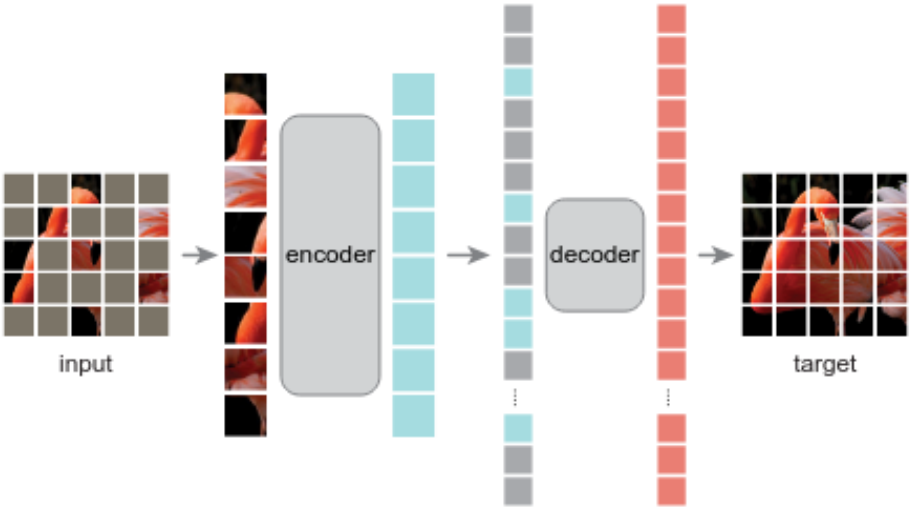
Θεωρητικό Υπόβαθρο

Σε αυτό το μέρος της εργασίας παρουσιάζουμε τις βασικές έννοιες και τεχνικές της βαθιάς μάθησης που αποτελούν το θεωρητικό υπόβαθρο της μεθόδου που προτείνουμε. Ξεκινάμε με μια επισκόπηση των βασικών παραδειγμάτων της μηχανικής μάθησης, εποπτευόμενη, μη εποπτευόμενη και αυτο-εποπτευόμενη μάθηση, δίνοντας ιδιαίτερη έμφαση στην τελευταία, καθώς η προσέγγισή μας βασίζεται εξ ολοκλήρου σε αυτο-εποπτευόμενες τεχνικές. Στη συνέχεια, περιγράφουμε τις βασικές αρχιτεκτονικές βαθιάς μάθησης που χρησιμοποιούνται στα πειράματά μας. Τέλος, εισάγουμε τα βασικά στοιχεία της ενισχυτικής μάθησης η οποία είναι κρίσιμη για την εκπαίδευση πολιτικών που επιτρέπουν στο σύστημά μας να αλληλεπιδρά έξυπνα με το περιβάλλον του.

Η εποπτευόμενη μάθηση είναι ένα υποπεδίο της μηχανικής μάθησης που βασίζεται σε επισημασμένα σύνολα δεδομένων για την εκπαίδευση μοντέλων με σκοπό την πρόβλεψη αποτελεσμάτων ή την αναγνώριση προτύπων. Τα μοντέλα μαθαίνουν να συσχετίζουν τα δεδομένα εισόδου με τις αντίστοιχες ετικέτες και έτσι μπορούν να γενικεύουν σε νέα, άγνωστα δεδομένα με υψηλή ακρίβεια [30]. Αντίθετα, **η μη εποπτευόμενη μάθηση** δεν χρησιμοποιεί ετικέτες. Στοχεύει στην ανακάλυψη δομών στα δεδομένα, στην κατηγοριοποίηση τους σε ομάδες (clusters) ή ακόμα στην εξαγωγή λανθάνωντων αναπαραστάσεων, χωρίς να υπάρχει άμεσο σήμα επίβλεψης.

Η αυτο-εποπτευόμενη μάθηση (Self-Supervised Learning, SSL) βρίσκεται στο μεταίχμιο αυτών των δύο προσεγγίσεων. Εκμεταλλεύεται τη δομή των δεδομένων για να δημιουργήσει ψευδο-ετικέτες, επιτρέποντας την εκπαίδευση μοντέλων σε εργασίες που παραδοσιακά απαιτούν επίβλεψη. Τώρα, τα σήματα επίβλεψης προκύπτουν από τα ίδια τα δεδομένα, επιτρέποντας στο μοντέλο να μάθει χρήσιμες αναπαραστάσεις χωρίς την ανάγκη ανθρώπινης επίβλεψης. Στην εικόνα 1, παρουσιάζεται μια ευρέως διαδεδομένη τεχνική SSL.

Η μέθοδος αυτή είναι πολύτιμη σε περιβάλλοντα όπου οι επισημειώσεις των δεδομένων απουσιάζουν ή ακόμα η συλλογή τους είναι χρονοβόρα και κοστοβόρα. Σε τέτοιες περιπτώσεις,



Σχήμα 1: Μια ευρέως διαδεδομένη τεχνική SSL[2]: Ένα μεγάλο μέρος της εικόνας εισόδου κρύβεται και το δίκτυο κωδικοποιεί την υπόλοιπη εικόνα. Μετά, ο αποκωδικοποιητής επιχειρεί να ανακατασκευάσει ολόκληρη την αρχική εικόνα.

μέσω της SSL επιτυγχάνεται η μάθηση απευθείας από ακατέργαστα δεδομένα. Επιπλέον, η αυτοεποπτευόμενη μάθηση προσεγγίζει περισσότερο τον τρόπο με τον οποίο οι βιολογικοί οργανισμοί αποκτούν γνώση. Συγκεκριμένα, η μάθηση στηρίζεται στην παρατήρηση και την πρόβλεψη - αξιοποιείται η εσωτερική δομή και η χρονική συνέπεια των δεδομένων για τη δημιουργία σημάτων μάθησης.

Όταν ο στόχος δεν είναι προκαθορισμένος και η κατανομή των δεδομένων είναι άγνωστη ή μεταβαλλόμενη, τα παραδοσιακά πλαίσια εποπτευόμενης ή μη εποπτευόμενης μάθησης αποτυγχάνουν. Σε τέτοιες συνθήκες νέα αντικείμενα, καταστάσεις ή στόχοι εμφανίζονται σε πραγματικό χρόνο. Μέσω της SSL μπορεί να επιτραπεί στους δράστες να μαθαίνουν συνεχώς μέσω αλληλεπίδρασης, αντιμετωπίζοντας κάθε εμπειρία ως πιθανό σήμα μάθησης, χωρίς την ανάγκη για ανθρώπινη παρέμβαση.

Συνήθεις αρχιτεκτονικές δικτύων βαθιάς μάθησης

Η βαθιά μάθηση έχει επιδείξει τεράστια επιτυχία τα τελευταία χρόνια σε τομείς όπως η όραση υπολογιστών, η επεξεργασία φυσικής γλώσσας και η ρομποτική. Σημαντικό μέρος αυτής της επιτυχίας οφείλεται στις αρχιτεκτονικές των μοντέλων που χρησιμοποιούνται οι οποίες είναι σχεδιασμένες έτσι ώστε να εκμεταλλεύονται τη δομή και τα χαρακτηριστικά των δεδομένων και να εξάγουν χρήσιμες αναπαραστάσεις απευθείας από τα δεδομένα [3]. Η πιο απλή τέτοια αρχιτεκτονική είναι το **Multilayer Perceptron (MLP)** το οποίο αποτελείται από πλήρως συνδεδεμένα επίπεδα. Σε κάθε επίπεδο εφαρμόζεται διαδοχικά ένας γραμμικός μετασχηματισμός και μια μη γραμμική συνάρτηση. Τέτοια μοντέλα, παρότι απλοϊκά επιδεικνύουν ανταγωνιστικές

επιδόσεις σε προβλήματα όπως η ταξινόμηση εικόνας [31].

Μια άλλη ιδιαιτέρως διαδεδομένη αρχιτεκτονική, ιδανική για την επεξεργασία δεδομένων που παρουσιάζουν τοπολογία πλέγματος είναι τα Συνελικτικά Νευρωνικά Δίκτυα, **Convolutional Neural Networks (CNNs)**. Στα CNNs η βασική πράξη είναι η συνέλιξη. Λόγω αυτού παρουσιάζουν σημαντικά χαρακτηριστικά όπως

1. **Αραιή Συνδεσιμότητα:** Οι συνδέσεις μεταξύ των διαδοχικών επιπέδων είναι αραιές αφού κατά την πράξη της συνέλιξης γίνεται χρήση μικρών φίλτρων. Έτσι τα δίκτυα αυτά έχουν λιγότερες παραμέτρους, μικρότερο υπολογιστικό κόστος, ενώ σέβονται τη χωρική τοπικότητα των δεδομένων
2. **Ισομεταβλητότητα ως προς τη χωρική μετατόπιση:** Μια μετατόπιση της εισόδου της πράξης της συνέλιξης έχει ως αποτέλεσμα αντίστοιχη μετατόπιση της εξόδου, ιδιότητα χρήσιμη σε προβλήματα αναγνώρισης εικόνας
3. **Διαμοιρασμός Παραμέτρων:** Ο ίδιος πυρήνας χρησιμοποιείται σε διαφορετικά χωρικά σημεία της εισόδου υποθέτοντας ότι χαρακτηριστικά χρήσιμα σε μία περιοχή της εικόνας θα είναι χρήσιμα και σε άλλες περιοχές

Τα CNNs είναι ακόμα εμπνευσμένα από τη βιολογία. Όπως και ο οπτικός φλοιός των θηλαστικών παρουσιάζουν ιεραρχική δομή [32], καθώς τα χαμηλότερα επίπεδα των δικτύων εξάγουν απλά χαρακτηριστικά όπως γωνίες και ακμές, ενώ τα υψηλότερα επίπεδα εξάγουν πιο σύνθετη συχνά σημασιολογική πληροφορία. Τα δίκτυα με υπολειμματικές συνδέσεις **ResNet**, [33], αποτελούν επίσης μια από τις πιο επιδραστικές αρχιτεκτονικές στη βαθιά μάθηση. Εισάγοντας τις υπολειμματικές αυτές συνδέσεις τα δίκτυα αυτά καταφέρνουν να αντιμετωπίσουν το πρόβλημα των εξαφανισμένων παραγώγων [34] που συναντάται στα βαθιά νευρωνικά δίκτυα. Μέσω των συνδέσεων αυτών δίνεται η δυνατότητα στο δίκτυο να παραλείπει ορισμένα επίπεδα, το οποίο οδηγεί στην αποτελεσματικότερη εκπαίδευση τους.

Vision Transformers: Παρά το γεγονός ότι η πράξη της συνέλιξης θεωρούνταν απαραίτητη για την αποτελεσματική επεξεργασία εικόνας οι Vision Transformers (ViT) [5] έρχονται να αμφισβητήσουν αυτήν την πεποίθηση. Οι Dosovitskiy et al. σπάνε μια εικονα σε τμήματα και την αντιμετωπίζουν ως μια ακολουθία. Στη συνέχεια, εφαρμόζουν κατάλληλους μηχανισμούς προσοχής [6] πάνω σε αυτήν την ακολουθία με αποτέλεσμα ακόμα και να ξεπερνούν τις επιδόσεις των Συνελικτικών Δικτύων.

Νευρωνικά Δίκτυα Γράφων (GNNs): Οι προηγούμενες αρχιτεκτονικές αφορούν κυρίως δεδομένα με ευκλείδεια δομή. Όταν τα δεδομένα είναι δομημένα ως γράφος, συχνά τα μοντέλα αυτά αποτυγχάνουν. Κοινωνικά δίκτυα, χημικά μόρια, προτάσεις φυσικής γλώσσας ή ακόμα και εικόνες μπορούν να μοντελοποιηθούν σαν ένας γράφος. Τα Νευρωνικά Δίκτυα Γράφων εφαρμόζονται πάνω σε τέτοιου είδους δεδομένα ώστε να προβλέπουν τα χαρακτηριστικά νέων κόμβων του γράφου, την ύπαρξη ή μη ακμών ή ακόμα και να ταξινομούν ολόκληρους νέους γράφους.

Η κύρια πράξη που εφαρμόζεται στους γράφους είναι το πέρασμα μηνύματος, message-passing, κατά το οποίο τα χαρακτηριστικά κάθε κόμβου του γράφου ενημερώνονται με βάση αυτά των γειτόνων του.

Η ενισχυτική μάθηση

Η ενισχυτική μάθηση (Reinforcement Learning - RL) είναι ένας ιδιαίτερος τρόπος μάθησης που βασίζεται στην αλληλεπίδραση ενός δράστη (agent) με το περιβάλλον του. Σε αντίθεση με άλλες μορφές μηχανικής μάθησης, στην ενισχυτική μάθηση ο δράστης μαθαίνει μόνος του, μέσα από την αλληλεπίδραση του με το περιβάλλον του και τη συνεχή αξιολόγηση των δράσεων του. Ο στόχος του είναι να επιλέγει δράσεις που οδηγούν στη μεγιστοποίηση της ανταμοιβής που λαμβάνει από το περιβάλλον. Αυτή η συνεχής διαδικασία μάθησης σχετίζεται με τον τρόπο που οι βιολογικοί οργανισμοι μαθαίνουν μέσω δοκιμής και σφάλματος.

Ένα σύστημα ενισχυτικής μάθησης αποτελείται από: τον δράστη, το περιβάλλον, μια πολιτική (δηλαδή τον τρόπο με τον οποίο συσχετίζονται οι καταστάστεις στις οποίες βρίσκεται ο δράστης με ενέργειες), ένα σήμα ανταμοιβής που καθορίζει τι θεωρείται επιτυχία και μια συνάρτηση αξίας που βοηθά τον δράστη να προβλέπει πόσο ωφέλιμη είναι μια κατάσταση με βάση τις μελλοντικές ανταμοιβές. Η διαδικασία ακολουθεί ένα συγκεκριμένο πλαίσιο, γνωστό ως Μαρκοβιανή Διαδικασία Απόφασης (Markov Decision Process), όπου κάθε απόφαση επηρεάζει όχι μόνο το παρόν αλλά και τις μελλοντικές καταστάσεις και ανταμοιβές. Ο δράστης καλείται να ισορροπήσει μεταξύ της εξερεύνησης νέων λιγότερο βέβαιων επιλογών και της εκμετάλλευσης όσων ήδη γνωρίζει ότι αποδίδουν καλά.

Η αξιολόγηση των ενεργειών γίνεται μέσω συναρτήσεων αξίας, οι οποίες αποτιμούν τη συνολική ανταμοιβή που μπορεί να περιμένει ο δράστης από μια κατάσταση ή από ένα συνδυασμό κατάστασης και ενέργειας. Η βέλτιστη στρατηγική μάθησης προκύπτει όταν ο δράστης μαθαίνει να επιλέγει ενέργειες που οδηγούν στη μέγιστη δυνατή αναμενόμενη ανταμοιβή. Αυτές οι έννοιες βρίσκονται στο επίκεντρο πολλών αλγορίθμων ενισχυτικής μάθησης, όπως και του αλγόριθμου Q-learning.

Το (Deep) **Q-learning** είναι μια μέθοδος ενισχυτικής μάθησης όπου ένας δράστης μαθαίνει να επιλέγει ενέργειες που οδηγούν στο μεγαλύτερο δυνατό όφελος. Χρησιμοποιεί ένα νευρωνικό δίκτυο για να εκτιμά, για κάθε κατάσταση, την αναμενόμενη αξία κάθε ενέργειας, ενώ προσεγγίζει το πραγματικό αναμενόμενο όφελος μιας κατάστασης αθροίζοντας το όφελος που λαμβάνει εμπειρικά και το όφελος που προβλέπει το δίκτυο ότι θα λαβει στην επόμενη κατάσταση που θα βρεθεί.

Για να βελτιωθεί η σταθερότητα της εκπαίδευσης, οι Mnih et al. [35], εφαρμόζουν τρεις τεχνικές: χρήσιμοποιούν ξεχωριστό δίκτυο για την προσέγγιση του αναμενόμενου όφελους, αποθηκεύουν παλαιότερες εμπειρίες σε μια μνήμη ώστε να τις επαναχρησιμοποιούν τακτικά και επιλέγουν ενέργειες με έναν τυχαίο αλλά ελεγχόμενο τρόπο (πολιτική ϵ-greedy). Η παραλλαγή του, **Double DQN** [36], προτείνει να διαχωρίζεται το δίκτυο που επιλέγει την ενέργεια από αυτό που την αξιολογεί, αποτρέποντας υπερβολικά αισιόδοξες εκτιμήσεις και βελτιώνοντας τη

σταθερότητα και την ακρίβεια της μάθησης.

Σχετική Βιοβλιογραφια

Στη συνέχεια παρουσιάζουμε τη βιβλιογραφία που σχετίζεται με τα κυριότερα θέματα που καταπιάνεται η παρούσα διπλωματική. Ξεκινάμε αναφερόμενοι στις βασικές μεθόδους κατάτμησης εικόνας σε αντικείμενα, αφού το πρόβλημα που έχουμε θέσει μπορεί αρχικά να αντιμετωπιστεί ως ένα πρόβλημα όρασης υπολογιστών, ενώ δίνουμε έμφαση σε αντικειμενο-κεντρικές μεθόδους κατάτμησης. Στη συνέχεια, εξετάζουμε τεχνικές μη επιβλεπομένης ενισχυτικής μάθησης αλλά και εσωτερικά παρακινούμενα σήματα επιβράβευσης. Αναλύοντας τη σχετική βιβλιογραφία θέτουμε τη βάση για την βαθύτερη κατανόηση της προτεινόμενης μεθοδολογίας.

Μη εποπτευόμενη κατάτμηση βίντεο Η αντικειμενο-κεντρική κατάτμηση (object segmentation) αναφέρεται στη διαδικασία κατά την οποία κάθε pixel μιας εικόνας ταξινομείται ως μέρος ενός συγκεκριμένου αντικειμένου ή του φόντου, με σκοπό τον ακριβή εντοπισμό των ορίων των αντικειμένων [37]. Στην περίπτωση των βίντεο, η αντίστοιχη διαδικασία ονομάζεται Video Object Segmentation (VOS) και στοχεύει στον εντοπισμό των σημαντικότερων αντικειμένων σε κάθε καρέ, σε επίπεδο pixel [37]. Η διαδικασία αυτή συνδέεται άμεσα με την οπτική αντίληψη, καθώς σύμφωνα με τον Biederman [38], η αναγνώριση αντικειμένων από τον άνθρωπο βασίζεται στη διάσπαση του οπτικου ερεθίσματος σε απλές γεωμετρικές μορφές. Στο πλαίσιο της μη εποπτευόμενης VOS, η δυσκολία αυξάνεται σημαντικά, αφού το μοντέλο πρέπει να εντοπίσει και να παρακολουθήσει τα αντικείμενα χωρίς τη χρήση επισημειωμένων δεδομένων ή αρχικών μασκών. Η ανάγκη για εντοπισμό και αναγνώριση αντικειμένων σε πολύπλοκες σκηνές καθιστά το πρόβλημα ιδιαίτερα απαιτητικό, και έχει οδηγήσει στην ανάπτυξη πληθώρας μεθόδων βασισμένων στη βαθιά μάθηση, καθώς και στη δημιουργία προτύπων συνόλων δεδομένων όπως το DAVIS [39] και το FBMS [40].

Αρχικές προσεγγίσεις, όπως αυτή των Fragkiadaki et al. [41], βασίστηκαν στην οπτική ροή για την ανίχνευση αντικειμένων και κίνησης, η οποία ενισχύεται μέσω συνελικτικών δικτύων. ΄Αλλες μέθοδοι, όπως αυτή των Zhou et al. [42], ενσωματώνουν μηχανισμούς χωρικής προσοχής και δυναμικής μέσω Convolutional LSTMs, ενώ πιο πρόσφατες προσεγγίσεις, όπως των Lu et al. [43], αξιοποιούν μηχανισμούς προσοχής για την ανίχνευση μακροπρόθεσμων εξαρτήσεων και τη βελτίωση της συνοχής των μασκών. Η μέθοδος των Yang et al. [12] συνδυάζει την οπτική ροή με αρχιτεκτονικές transformer, αντιμετωπίζοντας την κατάτμηση ως πρόβλημα ανακατασκευής της ροής. Τέλος, οι Lee et al. [44] προτείνουν ένα σύστημα με μια μορφή μνήμης, το οποίο επιτρέπει τη διατήρηση σημαντικών πληροφοριών από προηγούμενα καρέ, ενισχύοντας την χρονική συνέπεια και την ακρίβεια στην παρακολούθηση αντικειμένων σε βίντεο χωρίς εποπτεία.

Αντικειμενο-κεντρική κατάτμηση βίντεο Μια σημαντική κατεύθυνση στην προσπάθεια επίλυσης του προβλήματος της μη εποπτευόμενης κατάτμησης βιντεο είναι η αντικειμενο-κεντρική αναπαράσταση των σκηνών, όπου τα αντικείμενα θεωρούνται βασικές δομικές μονάδες. Οι περισσότερες σχετικές προσεγγίσεις στηρίζονται στην ανακατασκευή της εικόνας εισόδου, επιχειρώντας παράλληλα να απομονώσουν συνεκτικά συνδεδεμένα συστατικά της σκηνής. Οι μέθοδοι αυτές διακρίνονται κυρίως σε δύο κατηγορίες: τα μοντέλα μίξης σκηνής (scene-mixture models) και τα μοντέλα χωρικής προσοχής (spatial-attention models). Τα πρώτα, όπως τα MoNet [45], IODINE [46] και GENESIS [13], αντιμετωπίζουν τη σκηνή ως ένα συνδυασμό επιμέρους στοιχείων, τα οποία μοντελοποιούνται μέσω παραγωγικών μοντέλων. Ειδικότερα, το IODINE χρησιμοποιεί επαναληπτικά variational inference για την απομόνωση των αντικειμένων σε ανεξάρτητες λανθάνουσες μεταβλητές (slots), ενώ το GENESIS εισάγει σχέσεις χρονικής εξάρτησης μεταξύ των αντικειμένων αυτών. Ωστόσο, τα μοντέλα αυτά κωδικοποιούν έμμεσα τις χωρικές ιδιότητες των αντικειμένων στις λανθάνουσες μεταβλητές που αντιστοιχούν σε αυτά (όπως θέση και μέγεθος) και παρουσιάζουν προβλήματα κλιμάκωσης καθώς ο αριθμός αντικειμένων αυξάνεται.

Αντίθετα, τα μοντέλα χωρικής προσοχής [47, 48, 49, 50, 51] αναπαριστούν ρητά τις γεωμετρικές ιδιότητες των αντικειμένων και εστιάζουν στη διάκριση μεταξύ φόντου και προσκηνίου. Το κάθε αντικείμενο περιγράφεται μέσω χαρακτηριστικών όπως η θέση, η όψη και η παρουσία του. Το SCALOR [48], για παράδειγμα, παρουσιάζει βελτιόμενη πολυπλοκότητα επιτρέποντας παράλληλα τον εντοπισμό και την παρακολούθηση των αντικειμένων στο χρόνο. Παρά την υπολογιστική ευελιξία τους, τα μοντέλα αυτά συχνά αδυνατούν να αναπαραστήσουν σύνθετα σχήματα. Πρόσφατες προσεγγίσεις όπως το DINO-based object-centric attention [51] επιχειρούν να υπερβούν αυτούς τους περιορισμούς, αξιοποιώντας χάρτες προσοχής που προκύπτουν από αυτο-εποπτευόμενα μοντέλα μεγάλης κλίμακας. Σε αυτό το πλαίσιο, η μέθοδος Slot Attention [14] έχει αναδειχθεί ως ένα από τα πιο διαδεδομένα εργαλεία για τη μη εποπτευόμενη κατάτμηση και την μάθηση αναπαραστάσεων αντικειμένων και αποτελεί τη βάση της δικής μας προσέγγισης.

Μη εποπτευόμενη ενισχυτική μάθηση Η μη εποπτευόμενη ενισχυτική μάθηση (Unsupervised Reinforcement Learning) εστιάζει στην εκπαίδευση δραστών χωρίς την ύπαρξη εξωτερικού σήματος ανταμοιβής αλλά βασιζόμενη σε ενδογενή κίνητρα ή αυτο-εποπτευόμενα σήματα. Κλασικό παράδειγμα σε αυτό το πεδίο αποτελεί η εργασία Learning to Poke by Poking [52], όπου ένα ρομπότ μαθαίνει να προβλέπει την επίδραση των ενεργειών του πάνω σε αντικείμενα, αποκτώντας διαισθητική κατανόηση της φυσικής και της γεωμετρίας τους. ΄Αλλες εργασίες, όπως οι [18, 53], εφαρμόζουν off-policy βαθιά ενισχυτική μάθηση για την εκμάθηση στρατηγικών χειρισμού αντικειμενων, βασισμένες αποκλειστικά σε δεδομένα που συλλέγονται αυτόνομα. Σε αυτό το πλαίσιο, η προσέγγιση Grasp2Vec [54] συνδυάζει την εκμάθηση μιας πολιτικής χειρισμού αντικειμένων με την ταυτόχρονη εκμάθηση αναπαραστάσεων αντικειμένων.

Μια πιο πρόσφατη και ιδιαίτερα υποσχόμενη κατεύθυνση αφορά την αντικειμενο-κεντρική ενισχυτική μάθηση, όπου ο χώρος κατάστασης του δράστη οργανώνεται γύρω από οντότητες-αντικείμενα. Η προσέγγιση SMORL [19], για παράδειγμα, συνδυάζει αντικειμενο-κεντρική ανα-

παράσταση με ενισχυτική μάθηση δεδομένου ενός στόχου, χρησιμοποιώντας τον encoder SCALOR [48] για την παραγωγή δομημένων αναπαραστάσεων και έναν μηχανισμό προσοχής για την εκπαίδευση μιας πολιτικής βάσει στόχου. Η εκπαίδευση βασίζεται σε δεδομένα από τυχαίες τροχιές, ενώ ο δράστης θέτει δυναμικά στόχους που ορίζονται από εικόνες. Σε ανάλογο πνεύμα, οι Heravi et al. [55] αξιοποιούν την τεχνική του Slot Attention για την εξαγωγή αναπαραστάσεων αντικειμένων, δείχνοντας ότι η μοντελοποίηση των αντικειμένων οδηγεί σε σημαντικά καλύτερη απόδοση και ταχύτερη μάθηση, καθώς απαιτούνται λιγότερες δείγματα για την εκμάθηση της επιθυμητής πολιτικής.

Εσωτερικά παρακινούμενη ενισχυτική μάθηση Η εσωτερικά παρακινούμενη ενισχυτική μάθηση (Intrinsically Motivated Reinforcement Learning) αξιοποιεί εσωτερικούς μηχανισμούς επιβράβευσης για την καθοδήγηση της εξερεύνησης, ιδίως σε περιβάλλοντα όπου τα εξωτερικά σήματα επιβράβευσης είναι αραιά ή ακόμα δεν υπάρχουν. Σε αυτό το πλαίσιο, ο πράκτορας ενθαρρύνεται να επιλέγει ενέργειες που μειώνουν την αβεβαιότητά του και οδηγούν σε καταστάσεις με υψηλή πληροφοριακή αξία. Έργα όπως αυτά των Pathak et al. [56, 57] έδειξαν ότι η χρήση του σφάλματος πρόβλεψης ενός μοντέλου μπορεί να λειτουργήσει ως εγγενές σήμα ανταμοιβής, υπό την προϋπόθεση ότι οι αναπαραστάσεις εισόδου φιλτράρουν κατάλληλα λεπτομέρειες που δεν έχουν να κάνουν με τη δυναμική του περιβάλλοντος. Ωστόσο, η αποτελεσματικότητα τέτοιων μεθόδων περιορίζεται από την υψηλή διακύμανση και τη χαμηλή αποδοτικότητα δειγματοληψίας (sample efficiency) που χαρακτηρίζει τις παραδοσιακές τεχνικές ενισχυτικής μάθησης.

Για την αντιμετώπιση αυτών των περιορισμών, οι Pathak et al. [58] πρότειναν μια προσέγγιση βασισμένη στη διαφωνία μεταξύ ενός συνόλου από μοντέλα. Εκπαιδεύοντας κανείς ένα σύνολο (ensemble) μοντέλων μετάβασης, μπορεί να ορίσει το εγγενές σήμα ανταμοιβής ως την διακύμανση μεταξύ των προβλέψεών τους, προάγοντας την εξερεύνηση περιοχών του χώρου κατάστασης με υψηλή αβεβαιότητα. Επιπλέον, εναλλακτικές προσεγγίσεις, όπως αυτή των Liu et al. [59], βασίζονται στη μεγιστοποίηση της εντροπίας του χώρου κατάστασης, χρησιμοποιώντας εκτιμήτριες πυκνότητας του χώρου αυτού.

Η παρούσα διπλωματική εντάσσεται στο ευρύτερο πλαίσιο της ενεργής αντίληψης (active perception), όπου ο δράστης δεν περιορίζεται στην παθητική λήψη αισθητηριακών δεδομένων, αλλά αλληλεπιδρά ενεργά με το περιβάλλον για να βελτιώσει την αντιληπτική του ικανότητα. Εργασίες, όπως εκείνη των Pinto et al. [60], επιβεβαιώνουν ότι τεχνητοί δράστες μπορούν να αποκτήσουν οπτικές αναπαραστάσεις μέσω αλληλεπίδρασης, ενώ οι Pathak et al. [61] εισάγουν ένα αυτο-εποπτευόμενο σχήμα κατάτμησης αντικειμένων μέσω δοκιμών και παρατήρησης. Οι Sancaktar et al. [62] και Waters et al. [63] προτείνουν μεθόδους εγγενώς παρακινούμενης εξερεύνησης για τη συλλογή δεδομένων υψηλής πληροφορίας, αξιοποιώντας ensembles μοντέλων μετάβασης. Παρότι κοντά στη δική μας προσέγγιση, οι εν λόγω μέθοδοι είτε βασίζονται σε proprioceptive δεδομένα είτε αξιολογούνται σε απλοποιημένα περιβάλλοντα. Αντίθετα, η δική μας εργασία προτείνει, για πρώτη φορά, ένα πλήρως αυτο-εποπτευόμενο, αντικειμενο-κεντρικό σύστημα που ενισχύει εγγενώς την αντίληψη και τη μοντελοποίηση του κόσμου σε σύνθετα πε-

ριβάλλοντα.

Προτεινόμενη Μεθοδολογία

Παρουσιάζουμε τώρα την προτεινόμενη μεθοδολογία. Ξεκινάμε με μια επισκόπηση της συνολικής διαδικασίας και στη συνέχεια εξηγούμε λεπτομερώς κάθε επιμέρους στοιχείο. Αρχικά, περιγράφουμε πώς ένα μοντέλο όρασης μπορεί να εκπαιδευτεί ώστε να εκτελεί αντικειμενοκεντρική κατάτμηση χωρίς καθόλου επίβλεψη. Στη συνέχεια, παρουσιάζουμε ένα μοντέλο μετάβασης που εκπαιδεύεται να προβλέπει τη δυναμική αυτών των αναπαραστάσεων. Ακολουθεί η ανάλυση του εσωτερικού κινήτρου που χρησιμοποιούμε ως σήμα ανταμοιβής, καθώς και των αλγορίθμων ενισχυτικής μάθησης που αξιοποιούνται για την εκπαίδευση της πολιτικής. Τέλος, δείχνουμε πώς αυτή η πολιτική οδηγεί στη συλλογή πιο πλούσιων αλληλεπιδράσεων του δράστη με το περιβάλλον του, οι οποίες με τη σειρά τους βελτιώνουν περαιτέρω τα μοντέλα όρασης και μετάβασης.

Περίληψη
Ακολουθούμε μια αυτο-επιβλεπόμενη, αντικειμενο-κεντρική μέθοδο που αποτελείται από τέσσερα στάδια

α′) Ξεκινάμε συλλέγοντας ένα αρχικό σύνολο δεδομένων από ακολουθίες εικόνων που παράγει ένας δράστης μέσω τυχαίων κινήσεων στο περιβάλλον του.

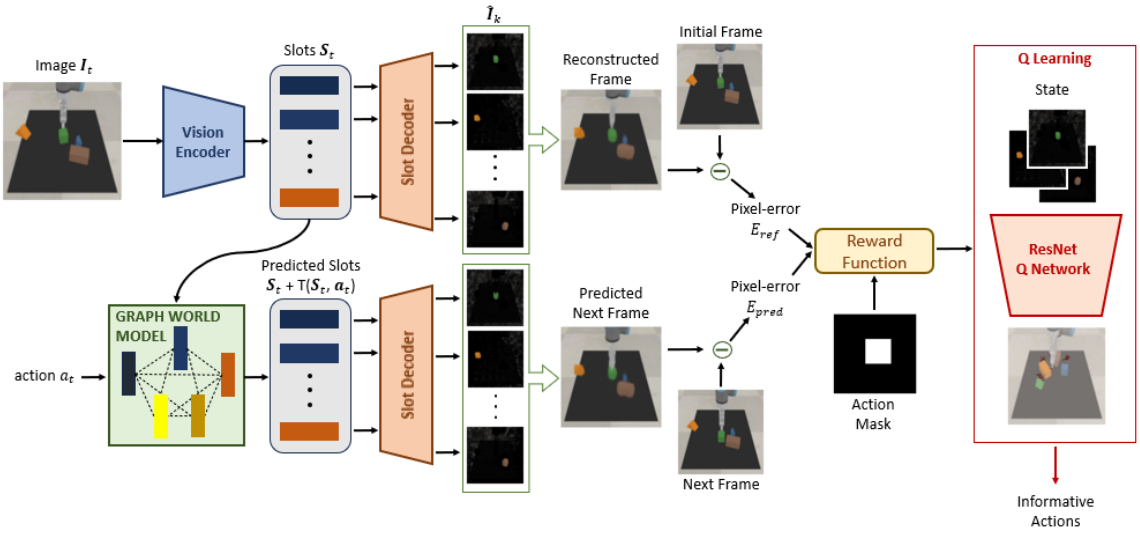
β′) Εκπαιδεύουμε ένα αυτο-επιβλεπόμενο μοντέλο όρασης βασισμένο στο Slot Attention, να κατατμήσει σκηνές σε τμήματα ενώ εξάγει αντικειμενο-κεντρικές αναπαραστάσεις.

γ′) Αξιοποιώντας το μοντέλο όρασης, εκπαιδεύουμε ένα μοντέλο μετάβασης να προβλέπει τη μελλοντική κατάσταση των οπτικών αναπαραστάσεων δεδομένου των ενεργειών του δράστη. Το μοντέλο μετάβασης αρχικά έχει περιορισμένες δυνατότητες εξαιτίας της μικρής διακύμανσης των δεδομένων που υπάρχουν στο αρχικό σύνολο δεδομένων.

δ′) Εκπαιδεύουμε μια πολιτική χρησιμοποιώντας ένα εσωτερικά παρακινούμενο σήμα επιβράβευσης η οποία ωθεί τον δράστη να εξερευνά καταστάσεις του περιβάλλοντος του για τις οποίες τα μοντέλα είναι αβέβαια.

ε′) Τελικά, χρησιμοποιούμε την νέα πολιτική για να συλλέξουμε ένα σύνολο τροχιών με περισσότερη πληροφορία για το περιβάλλον και να βελτιώσουμε τα μοντέλα όρασης και μετάβασης

Μια περίληψη της προτεινόμενης μεθοδολογίας φαίνεται στο σχήμα 2.



Σχήμα 2: **Περίληψη της προτεινόμενης μεθοδολογίας:** Η εικόνα εισόδου $\mathbf{I}_t$ επεξεργάζεται από έναν κωδικοποιητή και εξάγονται K slot representations που στη συνέχεια αποκωδικοποιούνται για να ανακατασκευαστεί η αρχική εικόνα $\hat{\mathbf{I}}_t$. Ένα μοντέλο μετάβασης βασισμένο σε γράφους παίρνει τα K slots και την αντίστοιχη δράση και προβλέπει την μελλοντική εικόνα $\mathbf{I}_{t+r}$. Ένα σήμα επιβράβευσης υπολογίζεται με βάση το σφάλμα πρόβλεψης, το οποίο οδηγεί ένα Q-Network να προτείνει πιο ενδιαφέρουσες δράσεις.

Οπτικές Αναπαραστάσεις με Slot Attention

Για την εκμάθηση αυτο-επιβλεπόμενων, αντικειμενο-κεντρικών οπτικών αναπαραστάσεων, χρησιμοποιούμε τη μέθοδο **Slot Attention** [14], μία σύγχρονη και αποδοτική τεχνική για την κατάτμηση σκηνής σε αντικείμενα χωρίς εποπτεία. Η προσέγγιση αυτή εισάγει μεταβλητές, γνωστές ως *slots*, οι οποίες συνδέονται με την εικόνα εισόδου μέσω ενός διαφορίσιμου μηχανισμού προσοχής, ώστε να απομονώνουν διακριτές περιοχές της σκηνής που αντιστοιχούν σε αντικείμενα.

Η βασική ιδέα είναι απλή: αποδομούμε την εικόνα σε slots και επιχειρούμε να την ανακατασκευάσουμε μέσω αυτών. Η διαδικασία αυτή περιλαμβάνει δύο βασικά συστατικά: έναν **Κωδικοποιητή (Vision Encoder)** και έναν **Αποκωδικοποιητή (Slot Decoder)**, οι οποίοι επιτρέπουν στο σύστημα να μάθει να αναπαριστά σκηνές με τρόπο αντικειμενο-κεντρικό, χωρίς επισημειωμένα δεδομένα.

Κωδικοποιητής (Vision Encoder) Για κάθε βίντεο με καρέ $\mathbf{I}_t$ στη χρονική στιγμή t, ο κωδικοποιητής παράγει K slots $\mathbf{S}_t \in \mathbb{R}^{K \times D}$, όπου K είναι ο αριθμός των slots και D η διαστατικότητα κάθε slot. Η εικόνα εισόδου επεξεργάζεται πρώτα από ένα βαθύ νευρωνικό δίκτυο (π.χ. ένα CNN), το οποίο εξάγει N χαρακτηριστικα διαστάσεων D_f, $\mathbf{h}_t = f_{enc}(\mathbf{I}_t) \in \mathbb{R}^{N \times D_f}$.

Το **Slot Attention** αντιστοιχίζει επαναληπτικά τα slots στα χαρακτηριστικά $\mathbf{h}_t$ μέσω ενός μηχανισμού προσοχής. Τα slots ανταγωνίζονται μεταξύ τους ώστε να εξηγήσουν διαφορετικές περιοχές της εικόνας και βελτιώνονται σταδιακά με τη βοήθεια γραμμικών μετασχηματισμών, ενός MLP και μιας μονάδας GRU [64]. Τα slots αρχικοποιούνται με τυχαίες τιμές και εξελίσσονται με στόχο το κάθε ένα να αναπαριστά ένα διαφορετικό αντικείμενο.

Ο μηχανισμός προσοχής βασίζεται στη χρήση *keys* που προέρχονται από τα χαρακτηριστικά $\mathbf{h}_t$, ενώ τα *queries* και *values* προέρχονται από τα τρέχοντα slots $\mathbf{S}_t$. Τα βάρη προσοχής που προκύπτουν χρησιμοποιούνται για την ενημέρωση των slots, οδηγώντας έτσι στη διαμόρφωση διακριτών αναπαραστάσεων για κάθε αντικείμενο της σκηνής.

Αποκωδικοποιητής (Slot Decoder) Κάθε slot αποκωδικοποιείται χρησιμοποιώντας έναν αποκωδικοποιητή τύπου *spatial broadcast* [65], με στόχο την ανακατασκευή της περιοχής της σκηνής που του αντιστοιχεί. Κάθε slot $\mathbf{s}_k$ παράγει μια εικόνα ανακατασκευής $\hat{\mathbf{I}}_k$ και μία μάσκα $\mathbf{\Pi}_k$, η οποία κανονικοποιείται μέσω ενός softmax. Η τελική εικόνα προκύπτει συνδυάζοντας τα επιμέρους στοιχεία:

$$\hat{\mathbf{I}}_k, \hat{\mathbf{\Pi}}_k = f_{dec}(\mathbf{s}_k), \quad \mathbf{\Pi}_k = softmax_K \hat{\mathbf{\Pi}}_k \quad \hat{\mathbf{I}} = \sum_{k=1}^{K} \mathbf{\Pi}_k \odot \hat{\mathbf{I}}_k. \tag{1}$$

Για ευκολία, γράφουμε συνοπτικά $\hat{\mathbf{I}}_t = f_{dec}(\mathbf{S}_t)$, όπου f_{dec} αντιπροσωπεύει ολόκληρη τη διαδικασία αποκωδικοποίησης.

Εκπαίδευση μέσω Ανακατασκευής Μπορούμε να εκπαιδεύσουμε από κοινού το CNN του κωδικοποιητή, το μηχανισμό Slot Attention, καθώς και τον αποκωδικοποιητή, απλώς μέσω της ανακατασκευής των αρχικών καρέ:

$$L_{rec} = \sum_{t=1}^{T} \|\hat{\mathbf{I}}_t - \mathbf{I}_t\|^2 \tag{2}$$

Τροποποιήσεις στο Slot Attention Παρόλο που η αντικειμενο-κεντρική όραση υπολογιστών παρουσιάζει σημαντική πρόοδο, ο τομέας βρίσκεται ακόμη σε πρώιμο στάδιο. Οι περισσότερες μέθοδοι βασίζονται σε ασθενώς εποπτευόμενα σήματα, όπως η οπτική ροή ή σε προεκπαιδευμένα μοντέλα μεγάλης κλίμακας. Εμείς εκπαιδεύουμε το Slot Attention **από το μηδέν**, χρησιμοποιώντας μόνο δεδομένα που συλλέγονται αυτόνομα από έναν ρομποτικό βραχίονα.

Για τη βελτίωση των αποτελεσμάτων, εφαρμόζουμε τις εξής τροποποιήσεις:

1. **Κωδικοποιητής ResNet:** Αντί για έναν κωδικοποιητή (Vision Transformer) που προεκπαιδεύεται συνήθως με τη μέθοδο DINO [66] σε ImageNet, χρησιμοποιούμε έναν απλό

κωδικοποιητή τύπου ResNet, ο οποίος αποδίδει καλύτερα στο δικό μας σύνολο δεδομένων.

2. **Αυτοεποπτευόμενη Προεκπαίδευση μέσω Αυτόματης Κωδικοποίησης (Autoencoder):** Για να επιταχύνουμε τη σύγκλιση και να σταθεροποιήσουμε την εκπαίδευση, προεκπαιδεύουμε τον κωδικοποιητή ResNet ως μέρος ενός Autoencoder. Το μοντέλο αποτελείται από τον κωδικοποιητή ResNet και έναν συμμετρικό αποκωδικοποιητή ResNet που βασίζεται σε αποσυνελίξεις (deconvolutions). Αυτή η μη εποπτευόμενη προεκπαίδευση επιτρέπει στο σύστημα να μάθει γενικά οπτικά χαρακτηριστικά χρήσιμα για την επακόλουθη αναπαράσταση αντικειμένων.

Η Εντροπία των μασκών ως όρος σφάλματος Εισάγουμε έναν επιπλέον όρο απώλειας που τιμωρεί την εντροπία των χωρικών μασκών $\mathbf{\Pi}_k$, ενισχύοντας τη συνοχή και μειώνοντας τον θόρυβο. Η τελική συνάρτηση σφάλματος γίνεται:

$$L = L_{rec} + \beta \cdot L_{ent} \tag{3}$$

όπου β είναι μία υπερπαράμετρος που ορίζεται σε 10^{-3}.

Από τις Εικόνες στα Βίντεο Για να εισάγουμε το χρόνο στο μοντέλο όρασης, χρησιμοποιούμε μια εκδοχή του Slot Attention που λειτουργεί διαδοχικά σε βίντεο. Αντί τα slots να αρχικοποιούνται τυχαία σε κάθε καρέ, χρησιμοποιείται ένα μεταβατικό μοντέλο που προβλέπει την επόμενη κατάσταση όπως στο SAVI [67].

Το μοντέλο μετάβασης

Ο στόχος μας είναι τώρα να εκπαιδεύσουμε ένα μοντέλο μετάβασης που μπορεί να προβλέπει την δυναμική των αντικειμένων, να προβλέπει δηλαδή την μελλοντική κατάσταση των slots δεδομένης μίας δράσης. Ακολουθούμε τους Kipf et al. [20] και χρησιμοποιούμε ένα πλήρως συνδεδεμένο Νευρωνικό Δίκτυο Γράφου που μαθαίνει βασιζόμενο σε τριάδες $(\mathbf{S}_t, a_t, \mathbf{S}_{t+r})$, όπου $\mathbf{S}_t$ είναι τα slots τη χρονική στιγμή t, a_t είναι η ενέργεια του δράστη και $\mathbf{S}_{t+r}$ τα μελλοντικά slots μετά από συγκεκριμένο χρονικό διάστημα r, αφού έχει ολοκληρωθεί η ενέργεια του δράστη. Το μοντέλο προβλέπει μεταβάσεις έτσι ώστε $\mathbf{S}_t + T(\mathbf{S}_t, a_t) \approx \mathbf{S}_{t+r}$.

Μπορούμε να εκπαιδεύσουμε το μοντέλο χρησιμοποιώντας το Μέσο Τετραγωνικό Σφάλμα στα slots:

$$L_{pred} = \|\mathbf{S}_t + T(\mathbf{S}_t, a_t) - \mathbf{S}_{t+r}\|_2 \tag{4}$$

Χρησιμοποιούμε ακόμα ενα σφάλμα αντίθεσης (contrastive loss), όπου συγκρίνουμε τα slots που προβλέπει το μοντέλο με τυχαία αλλοιωμένα slots $\mathbf{S}^c$:

$$L_{hinge} = max(0, \gamma - \|\mathbf{S}_t + T(\mathbf{S}_t, a_t) - \mathbf{S}^c\|_2) \tag{5}$$

Τέλος εισάγουμε ένα τρίτο όρο, ένα σφάλμα ανακατασκευής στο χώρο της εικόνας ώστε να επιβάλλουμε τα προβλεπόμενα slots να μπορούν να οπτικοποιηθούν σε πραγματικές αλλάγες

στο περιβάλλον του δράστη:

$$L_{rec} = \|f_{dec}(\mathbf{S}_t + T(\mathbf{S}_t, a_t)) - \mathbf{I}_{t+r}\|_2 \tag{6}$$

Τελικά, το σφάλμα εκπαίδευσης ορίζεται ως $L_{wm} = L_{pred} + L_{hinge} + \alpha L_{rec}$. Πρακτικά, χρησιμοποιούμε $\gamma = 10$ και $\alpha = 10^3$.

Σχεδίαση του εσωτερικά παρακινούμενου σήματος επιβράβευσης
Σχεδιάζουμε τώρα το εσωτερικά παρακινούμενο σήμα επιβράβευσης. Εξετάζουμε διαφορετικά σήματα ως σήματα επιβράβευσης. Αρχικά εξετάζουμε το σφάλμα ανακατασκευής του μοντέλου όρασης καθώς θεωρούμε ότι μεγάλο σφάλμα ανακατασκευής συνεπάγεται ότι το περιβάλλον του δράστη βρίσκεται σε μια πρωτόγνωρη κατάσταση. Εξετάζουμε ακόμα το σφάλμα πρόβλεψης του μοντέλου μετάβασης. Μια δράση που δεν μπορεί να προβλέψει το μοντέλο μετάβασης είναι χρήσιμη και μπορεί να οδηγήσει σε καλύτερη εξερεύνηση του περιβάλλοντος. Κοιτάζουμε ακόμα τη διαφωνία μεταξύ ενός συνόλου από μοντέλα μετάβασης. Αν τα μοντέλα μετάβασης παρουσιάζουν υψηλή διακύμανση ως προς τη πρόβλεψη του αποτελέσματος μια δράσης σημαίνει ότι υπάρχει αβεβαιότητα ως προς το αποτέλεσμα της δράσης και θα είναι χρήσιμο να επιλεγεί.

Αφού αναλύσουμε και συγκρίνουμε τα διαφορετικά αυτά σήματα σχεδιάζουμε ένα σήμα που βασίζεται στο σφάλμα πρόβλεψης του μοντέλου μετάβασης. Αρχικά, υπολογίζουμε το σφάλμα πρόβλεψης στο πεδίο της εικόνας.

$$\mathbf{E}_{pred} = (f_{dec}(\mathbf{S}_t + T(\mathbf{S}_t, a_t)) - \mathbf{I}_{t+r})^2 \tag{7}$$

Επειδή αυτό το σφάλμα συμπεριλαμβάνει και τα σφάλματα λόγω του μοντέλου όρασης είναι αρκετά ασταθές. Εισάγουμε ένα σφάλμα αναφοράς με το οποίο επιδιώκουμε να απομονώσουμε το σφάλμα που οφείλεται σε θορυβώδη οπτική αναπαράσταση:

$$\mathbf{E}_{ref} = (f_{dec}(\mathbf{S}_t) - \mathbf{I}_t)^2 \tag{8}$$

Τέλος, κάνουμε χρήση μιας διαδικής μάσκας γύρω από την περιοχή που πραγματοποιήθηκε η δράση. Τελικά το σήμα επιβράβευσης λαμβάνει τη μορφή:

$$r_t = \sum_i \sum_j [\mathbf{M} \odot max((\mathbf{E}_{pred} - \mathbf{E}_{ref}), 0)]_{ij} \tag{9}$$

Εκπαίδευση νέας πολιτικής Εκπαιδεύουμε τώρα μια πολιτική ώστε να επιλέγει κινήσεις που μεγιστοποιούν το σήμα που αναλύθηκε νωρίτερα. Για να το πετύχουμε αυτό, διατυπώνουμε το πρόβλημα ως μια διαδικασία απόφασης Markov (MDP), όπου η κατάσταση του συστήματος προκύπτει από τις 10 μάσκες που παράγονται κατά την αποκωδικοποίηση των slots και οι ενέργειες αντιστοιχούν σε κινήσεις του ρομποτικού βραχίονα. Χρησιμοποιούμε τον αλγόριθμο Double Q-learning με ένα δίκτυο ResNet, το οποίο δέχεται τις μάσκες ως είσοδο και προβλέπει σε ποιες περιοχές της εικόνας αξίζει να δράσει το ρομπότ. Κάθε ενέργεια είναι μια ώθηση σε κάποιο σημείο της εικόνας, η οποία μεταφράζεται σε μία κίνηση του βραχίονα.

Βελτίωση των μοντέλων Αφού εκπαιδευτεί η πολιτική, τη χρησιμοποιούμε για να συλλέξουμε νέα δεδομένα από το περιβάλλον. Αυτές οι αλληλεπιδράσεις οδηγούν σε καταστάσεις που δυσκολεύουν το μοντέλο μετάβασης και προσφέρουν στο μοντέλο όρασης νέες πληροφορίες. Με αυτόν τον τρόπο, βελτιώνουμε και τα δύο μοντέλα, εκπαιδεύοντάς τα πάνω σε δεδομένα που περιέχουν περισσότερες αλληλεπιδράσεις με αντικείμενα. Η διαδικασία αυτή δημιουργεί έναν κύκλο μάθησης, όπου το ρομπότ βελτιώνει συνεχώς την κατανόηση του κόσμου του χωρίς εξωτερική επίβλεψη.

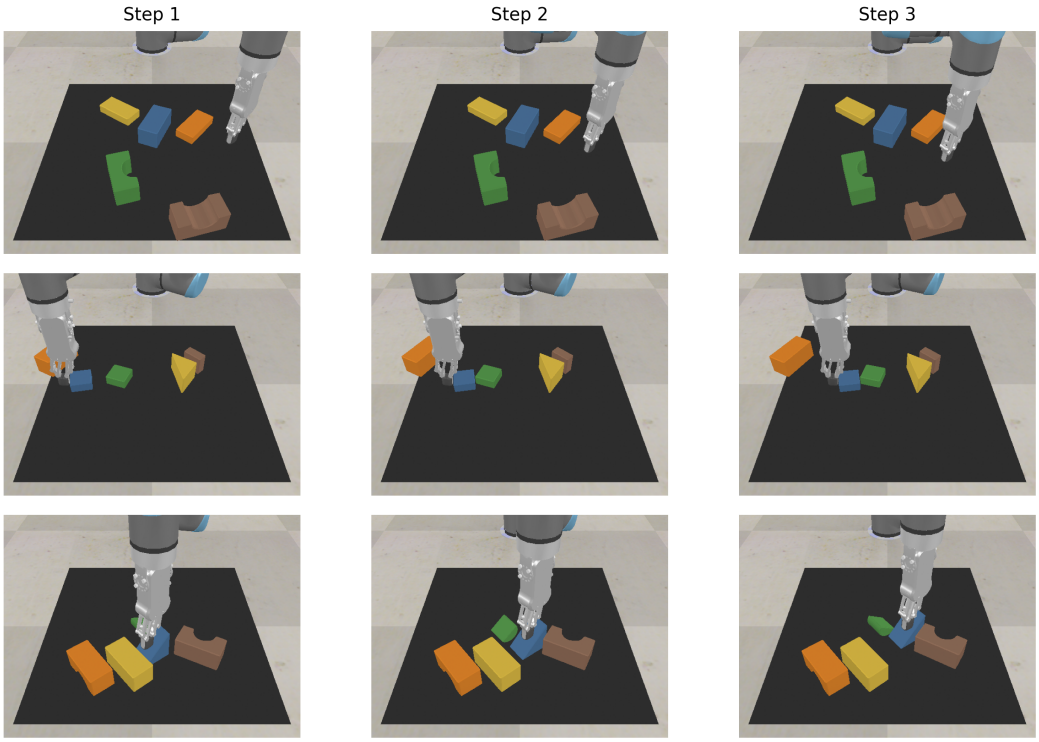
Πειραματικό Μέρος

Για να επαληθεύσουμε τη μεθοδολογία που προτείνουμε σχεδιάζουμε ένα κατάλληλο πειραματικό περιβάλλον χρησιμοποιώντας την προσομοίωση CoppeliaSim [68]. Το πείραμα περιλαμβάνει ένα ρομποτικό βραχίονα UR5, να αλληλεπιδρά με απλά τρισδιάστατα αντικείμενα σε ένα τραπέζι. Ο βραχίονας εκτελεί προκαθορισμένες κινήσεις σπρώχνοντας στο χώρο μπροστά του με συγκεκριμένη κατεύθυνση. Στη σκηνή υπάρχει κάμερα RGB-D, ενώ εφαρμόζονται μετασχηματισμοί συντεταγμένων μεταξύ της εικόνας και του φυσικού χώρου του ρομπότ, ώστε να επιτρέπεται η σύνδεση των ενεργειών του ρομπότ με οπτικά δεδομένα και αντίστροφα. Ο στόχος του συστήματος είναι να ανιχνευθούν τα αντικείμενα και να εξαχθεί πληροφορία για τα χαρακτηριστικά τους χωρίς κανένα σήμα επίβλεψης ή πρότερη γνώση. Όλα τα μοντέλα εκπαιδεύονται από το μηδέν, πάνω σε αλληλεπιδράσεις που έχει συλλέξει ο βραχίονας.

Αρχικά, ο βραχίονας αλληλεπιδρά τυχαία με το περιβάλλον διαλέγοντας από τις εφικτές κινήσεις. Σε αυτή τη φάση, φτιάχνουμε ένα σύνολο δεδομένων που αποτελείται από ζευγάρια δράσεων και ακολουθίες εικόνων, το οποίο καλούμε αρχικό σύνολο δεδομένων και το χρησιμοποιούμε για την αρχική εκπαίδευση των μοντέλων όρασης και μετάβασης. Φτιάχνουμε ακόμα ένα σύνολο δεδομένων ελέγχου στο οποίο ο βραχίονας είτε αλληλεπιδρά τυχαία με το περιβάλλον είτε χρησιμοποιώντας ευριστικές ώστε να εξασφαλίζεται ότι θα προκαλέσει κίνηση στα αντικείμενα του περιβάλλοντος. Στην εικόνα 3 δείχνουμε κάποια παραδείγματα των συνόλων δεδομένων.

Εκπαίδευση οπτικού μοντέλου και αρχιτεκτονικές

Αρχικά, η εκπαίδευση του μοντέλου όρασης γίνεται πάνω στο αρχικό σύνολο δεδομένων. Το μοντέλο Σλοτ Αττεντιον εκπαιδεύεται να διαχωρίζει την εικόνα σε επιμέρους αντικειμενοκεντρικές περιοχές και να ανακατασκευάζει τα καρέ, όπως φαίνεται στην εικόνα 4. Δοκιμάζονται τρεις διαφορετικές αρχιτεκτονικές για την εξαγωγή οπτικών χαρακτηριστικών. Η πρώτη, ένα απλό CNN, αποτυγχάνει να δώσει χρήσιμες αναπαραστάσεις, καθώς όλα τα slots καταλήγουν να περιγράφουν το φόντο.

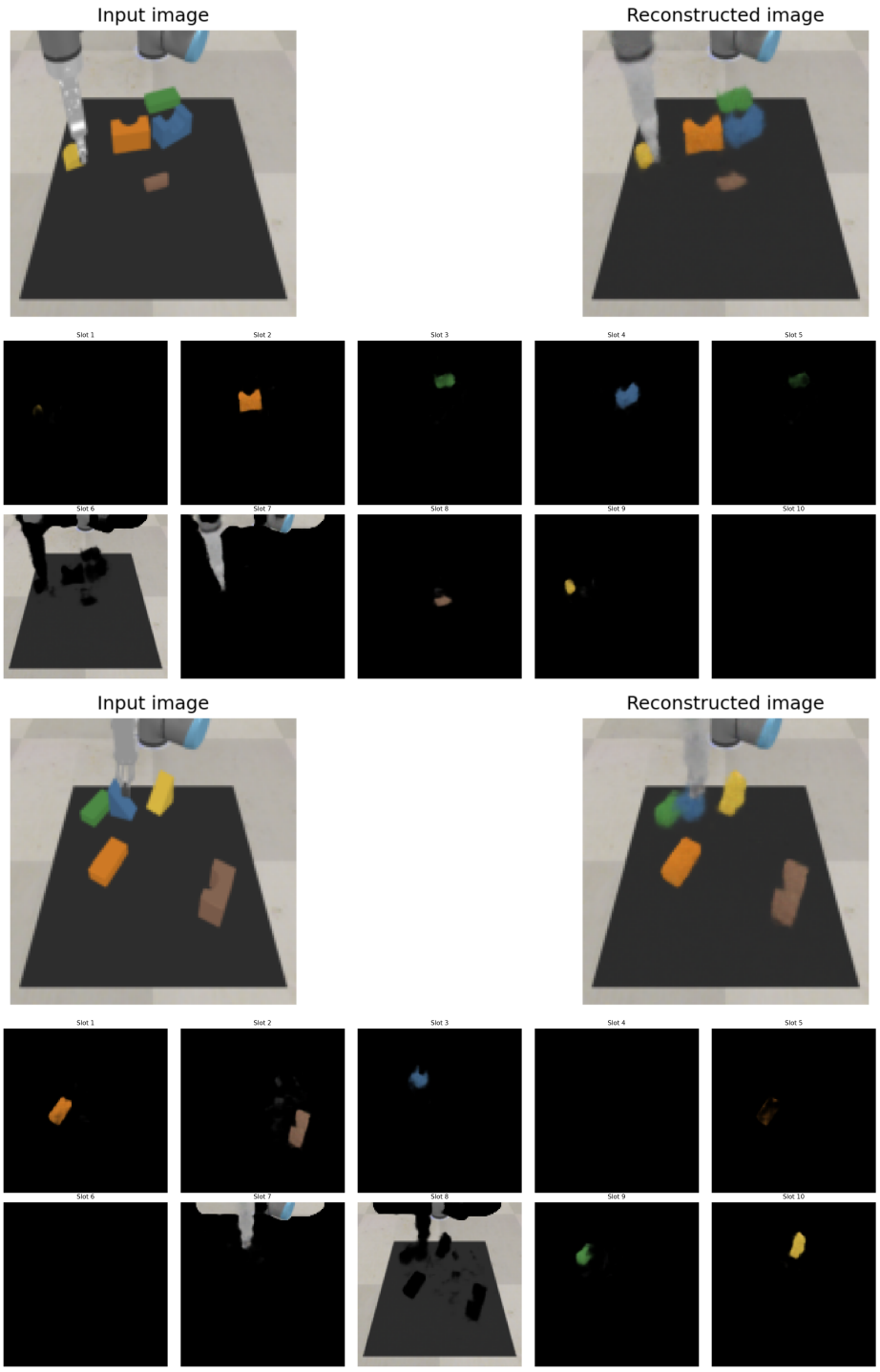


Σχήμα 3: **Πειραματικό Μέρος:** Ο ρομποτικός βραχίονας εκτελεί δράσεις στο περιβάλλον του

Η δεύτερη αρχιτεκτονική χρησιμοποιεί έναν προεκπαιδευμένο Vision Transformer (ViT). Αν και λειτουργεί αποτελεσματικά, η απόδοσή του περιορίζεται λόγω του μικρού μεγέθους του διαθέσιμου συνόλου δεδομένων. Τέλος, η τρίτη προσέγγιση βασίζεται σε έναν ResNet κωδικοποιητή που εκπαιδεύεται από την αρχή, χωρίς εξωτερική εποπτεία. Αυτή η προσέγγιση αποδεικνύεται ιδιαίτερα αποτελεσματική, ειδικά σε σενάρια με περιορισμένα δεδομένα, επιτυγχάνοντας υψηλής ποιότητας αναπαραστάσεις και αξιόπιστες ανακατασκευές. Στον πίνακα 1 παρουσιάζουμε μια ποσοτική σύγκριση των αρχιτεκτονικών.

Αρχιτεκτονική Κωδικοποιητή	**Δεδομένα Επαλήθευσης**
Απλό ‘‘NN	0.4
Προεκπαιδευμένος ΐT	0.095
ΡεσΝετ18	**0.073**

Πίνακας 1: **Σύγκριση μεταξύ διαφορετικών αρχιτεκτονικών κωδικοποιητή** Σημειώνουμε το σφάλμα ανακατασευής (ΜΣΕ $\xi 10^{-2}$) για τρεις αρχιτεκτονικές.

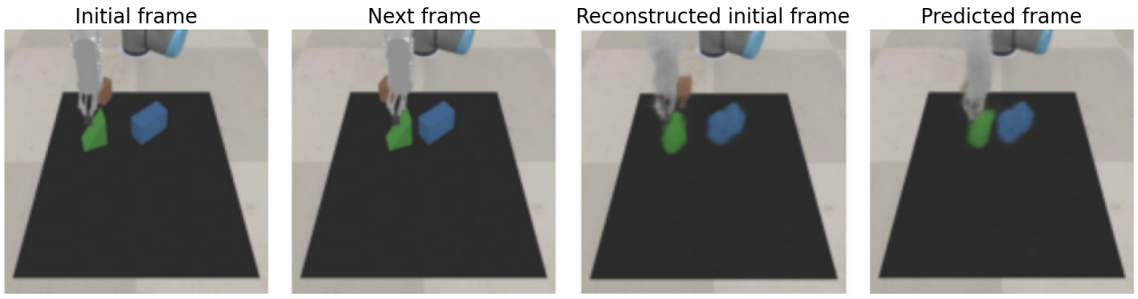


Σχήμα 4: Εκπαίδευση του μοντέλου όρασης: Το μοντέλο μαθαίνει να αντιστοιχίζει slots σε αντικείμενα ενώ ανακατασκυάζει ικανοποιητικά την εικόνα εισόδου.

Κανονικοποίηση των μασκών μέσω του σφάλματος εντροπίας Στην αρχή της εκπαίδευσης, οι μάσκες που παράγει το Slot Attention είναι αρκετά θορυβώδεις, αφού πολλά slots επεξηγούν κοινά σημεία της αρχικής εικόνας. Με τη χρήση της εντροπίας των μασκών ως σφάλμα ωθούμε τις μάσκες να έχουν τιμές κοντά στο 0 ή στο 1 με αποτέλεσμα καλύτερης ποιότητας κατάτμηση της εικόνας

Εκπαίδευση του μοντέλου μετάβασης Το μοντέλο μετάβασης εκπαιδεύεται ώστε να κατανοεί τις δυναμικές των αντικειμένων στη σκηνή, με βάση τις αναπαραστάσεις που προσφέρει το οπτικό μοντέλο. Για την εκπαίδευσή του, διατηρούμε παγωμένο το δίκτυο όρασης και χρησιμοποιούμε τα slots ως είσοδο. Κάθε αντικείμενο αναπαρίσταται ως ένα slot και κάθε δράση (π.χ. ώθηση του ρομπότ) αφού δεχθεί την επεξεργασία ενός MLP συνενώνεται με τις αναπαραστάσεις των αντικειμένων.

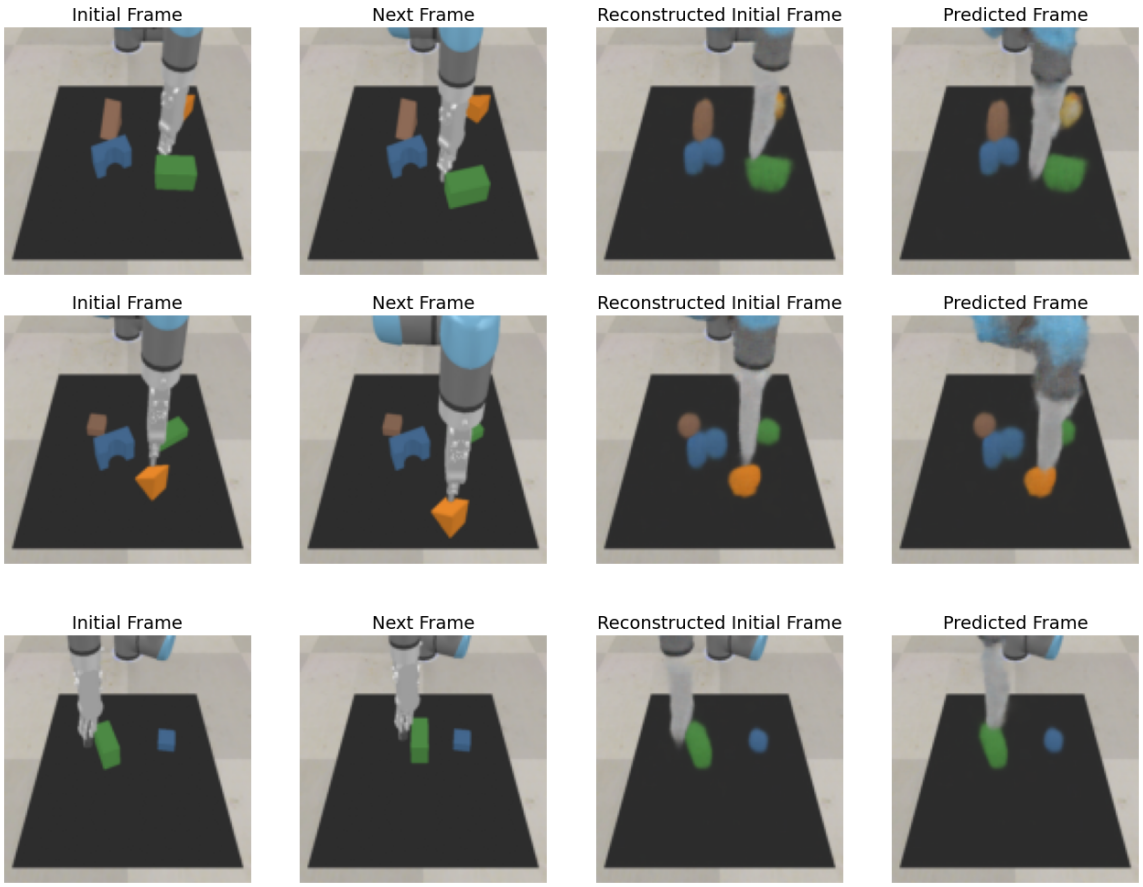
Η αρχιτεκτονική του μοντέλου βασίζεται σε ένα νευρωνικό δίκτυο γράφου (GNN), το οποίο μοντελοποιεί τις αλληλεπιδράσεις μεταξύ αντικειμένων. Μέσω κατάλληλου σχήματος message-passing μεταξύ κόμβων (αντικειμένων) το μοντέλο μαθαίνει να προβλέπει την επόμενη κατάσταση των αντικειμένων. Όταν εκπαιδεύεται με δεδομένα από δράσεις που προκαλούν κίνηση των αντικειμένων, το μοντέλο μπορεί να προβλέψει επιτυχώς την κίνηση τους, όπως φαίνεται στην εικόνα 5. Αντίθετα, όταν η εκπαίδευση βασίζεται σε τυχαίες δράσεις, το μοντέλο εστιάζει μόνο στην κίνηση του ίδιου του βραχίονα και αδυνατεί να αποδώσει σωστά τις δυναμικές των αντικειμένων, εικόνα 6.



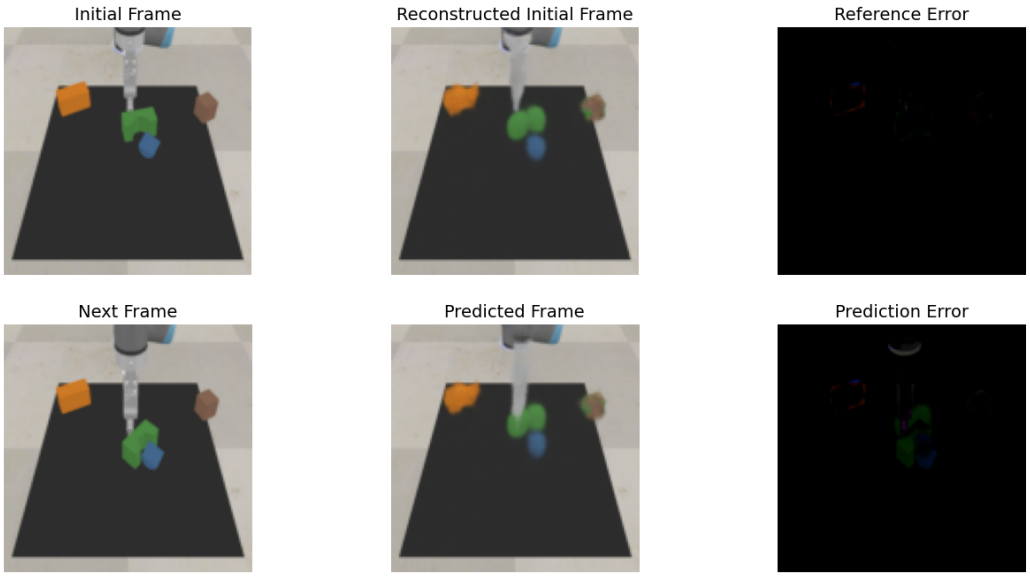
Σχήμα 5: Το μοντέλο μετάβασης είναι σε θέση να προβλέψει την κίνηση του πράσινου αντικειμένου, όταν έχει εκπαιδευτεί στο σύνολο δεδομένων με ευριστικές κινήσεις.

Σχεδίαση εσωτερικά παρακινούμενου σήματος ανταμοιβής Αφού το σύστημα έχει αποκτήσει μια λειτουργική κατανόηση του περιβάλλοντος, μπορούμε πλέον να καθοδηγήσουμε τη συλλογή πιο χρήσιμων δεδομένων. Το μοντέλο όρασης έχει εκπαιδευτεί επαρκώς ώστε να εντοπίζει αντικείμενα, ενώ το μοντέλο μετάβασης καταφέρνει να εκτιμά με σχετική ακρίβεια την κίνηση του ρομποτικού βραχίονα. Αυτές οι ικανότητες αρκούν για να εκπαιδεύσουμε μια πολιτική που επιδιώκει ενδιαφέρουσες τροχιές, δηλαδή ενέργειες που οδηγούν σε αλληλεπιδράσεις με τα αντικείμενα του περιβάλλοντος.

Προκειμένου να εντοπιστεί ένα κατάλληλο σήμα ανταμοιβής, αναλύονται τρεις διαφορετι-



Σχήμα 6: Όταν εκπαιδεύεται σε τυχαίες δράσεις το μοτέλο μετάβασης μπορεί μόνο να προβλέψει την κίνηση του βραχίονα.

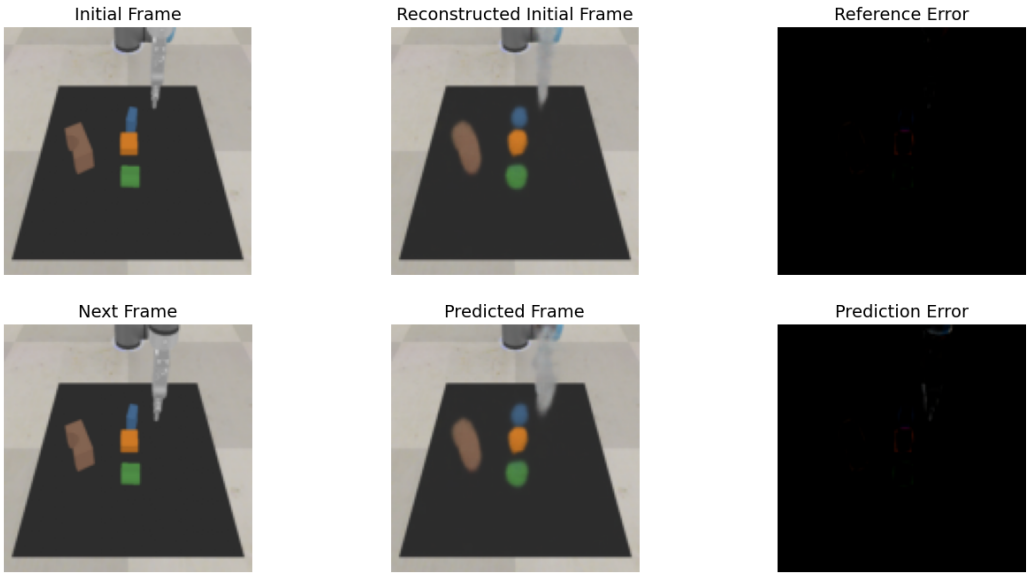


Σχήμα 7: Σχεδίαση του σήματος επιβράβευσης. Το μοντέλο μετάβασης παρουσιάζει υψηλό σφάλμα πρόβλεψης στην περιοχή που κινείται το πράσινο αντικείμενο. Αφαιρώντας το σήμα αναφοράς απομονώνουμε το σφάλμα που αφορά τη δράση του βραχίονα από το σφάλμα που αφορά την ανακατασκευή του πορτοκαλί αντικειμένου.

κές επιλογές: (i) το σφάλμα ανακατασκευής του μοντέλου όρασης, (ii) το σφάλμα πρόβλεψης του μοντέλου μετάβασης και (iii) η διακύμανση στις προβλέψεις ενός συνόλου μοντέλων μετάβασης. Αξιολογούμε τα σήματα κάτω από δύο διαφορετικά είδη δράσεων: τυχαίων και ευριστικών. Από τη σύγκριση διαπιστώνεται ότι τόσο το σφάλμα ανακατασκευής όσο και το σφάλμα πρόβλεψης αυξάνονται σημαντικά σε ενέργειες που προκαλούν κίνηση των αντικειμένων, γεγονός που τα καθιστά υποσχόμενα σήματα ανταμοιβής. Αντιθέτως, η διασπορά στις προβλέψεις του συνόλου μοντέλων δεν παρουσιάζει διαφοροποίηση μεταξύ των δράσεων διαφορετικού είδους και θεωρείται λιγότερο αξιόπιστη.

Ωστόσο, εμπειρικά παρατηρούμε ότι με χρήση των παραπάνω σημάτων δεν μπορούμε να εκπαιδεύσουμε πολιτικές, καθώς οι ανταμοιβές είναι συχνά ασταθείς ή θορυβώδεις. Για αυτό τον λόγο, προτείνουμε την ενίσχυση του σφάλματος πρόβλεψης με δύο επιπλέον μηχανισμούς όπως αναλύσαμε νωρίτερα. Στις εικόνες 7, 8 δείχνουμε τη διαίσθηση πίσω από τη σχεδίαση του σήματος που προτείνουμε. Με το σήμα αυτό καταφέρνουμε να εκπαιδεύσουμε μια πολιτική, ένα ResNet Q-Network, να προτείνει δράσεις που οδηγούν σε έως και τρεις φορές περισσότερη μετατόπιση των αντικειμένων του χώρου σε σύγκριση με τις τυχαίες δράσεις. Δείχνουμε τη σαφή προτίμηση της νέας πολιτικής να σπρώχνει τα αντικείμενα στην εικόνα 9.

Βελτίωση της Αντίληψης και της Πρόβλεψης του Συστήματος Αξιοποιώντας την πολιτική που εκπαιδεύσαμε, συλλέγουμε τώρα ένα νέο σύνολο δεδομένων. Με βάση αυτό το νέο

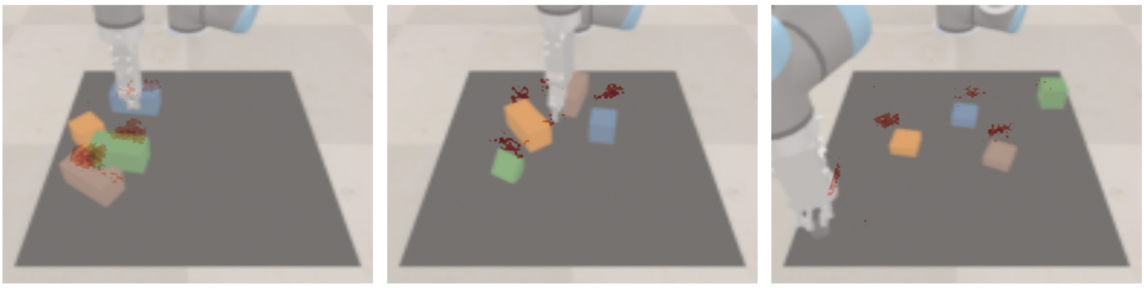


Σχήμα 8: Στις εικόνες χώρις κίνηση αντικειμένων στο σφάλμα πρόβλεψης κυριαρχεί το σφάλμα που οφείλεται στην ανακατασκεύη της εικόνας. Αφαιρώντας το σήμα αναφοράς καταφέρνουμε ο θόρυβος αυτός να μην συνεισφέρει στο τελικό σήμα επιβράβευσης με αποτέλεσμα την ελάχιστη επιβράβευση όταν υπάρχει απουσία κίνησης.

Εκπαίδευση σε:	**Αρχικά Δεδομένα**	**Ευριστικα Δεδομένα**
Τυχαίες Δράσεις	**0.062**	0.084
Τυχαίες Δράσεις	0.070	**0.053**
Δράσεις από την πολιτική μας	0.063	0.069

Πίνακας 2: **Αξιολόγηση του μοντέλου όρασης σε διαφορετικά δεδομένα** Σημειώνουμε το σφάλμα ανακατασκεύης (MSE x10^{-2}) του μοντέλου όρασης όταν εκπαιδεύεται σε τυχαίες δράσεις, σε ευριστικές δράσεις και στην εσωτερικά παρακινούμενη πολιτική μας.

υλικό, επανεκπαιδεύουμε τόσο το μοντέλο όρασης όσο και το μοντέλο κόσμου, ξεκινώντας από τις αρχικές τους παραμέτρους. Η επανεκπαίδευση αυτή οδηγεί σε σημαντική βελτίωση: το μοντέλο όρασης καταφέρνει πλέον να ανακατασκευάζει καλύτερα τις σκηνές, ενώ το μοντέλο κόσμου προβλέπει με μεγαλύτερη ακρίβεια τις επιπτώσεις των δράσεων όπως αναδεικνύεται στους πίνακες 2, 3. Συγκεκριμένα, στην τέταρτη στήλη της εικόνας 10 βλέπουμε ότι το μοντέλο μετάβασης είναι σε θέση πια να προβλέψει την κίνηση του αντικειμένου που σπρώχνεται από τον βραχίονα, όπως φαίνεται και ποσοτικά στον πίνακα 5.3. Εξετάζουμε δύο τρόπους βελτίωσης των μοντέλων: 1) διατηρούμε παγωμένο τον αρχικό κωδικοποιητή όρασης και επανεκπαιδεύου-



Σχήμα 9: **Οπτικοποίηση της πολιτικής:** Οπτικοποιούμε την έξοδο του Q-network. Δείχνουμε με κόκκινο τα σημεία τα οποία προτείνει η πολιτική που έχουμε εκπαιδεύσει. Φανερά, οι δράσεις είναι τέτοιες ώστε να προκαλείται κίνηση των αντικειμένων.

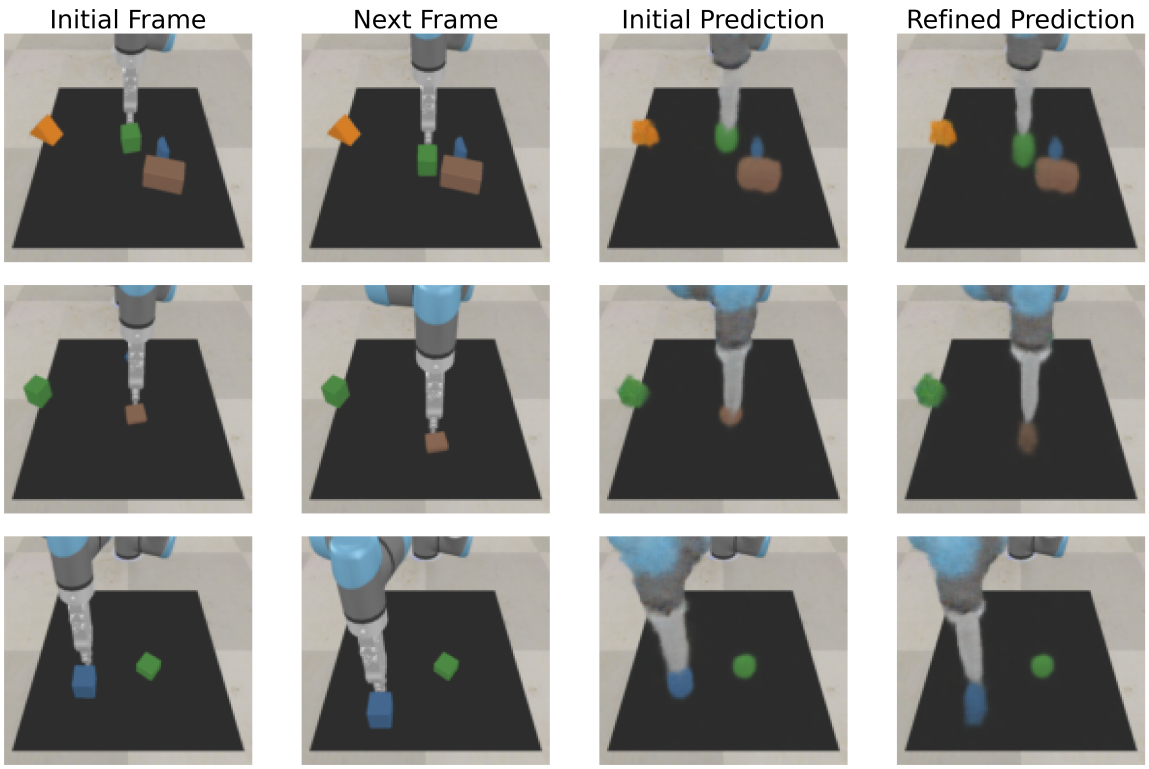
Εκπαίδευση σε:	**Αρχικά Δεδομένα**	**Ευριστικα Δεδομένα**
Τυχαίες Δράσεις	0.209	0.235
Ευριστικές Δράσεις	0.247	0.164
Εσωτερικά Παρακινούμενη Πολιτική	**0.131**	0.143
Εσωτερικά Παρακινούμενη Πολιτική(EV)	**0.131**	**0.135**

Πίνακας 3: **Αξιολόγηση του μοντέλου μετάβασης σε διαφορετικά σύνολα δεδομένων** Σημειώνουμε το σφάλμα ανακατσκευής (MSE $\text{x}10^{-2}$) κατά την πρόβλεψη του επόμενου καρέ και συγκρίνουμε 4 περιπτώσεις. Όταν το μοντέλο εκπαιδεύεται μόνο σε τυχαίες δράσεις, μόνο σε ευριστικές δράσεις ή όταν εκπαιδεύεται στα δεδομένα που παράγει η πολιτική που προτείνουμε. Στην τελευταία περίπτωση (EV) εξετάζουμε και την δυνατότητα χρήσης του μοντέλου όρασης που έχει εκπαιδευτεί και στα νέα δεδομένα.

με μόνο το μοντέλο μετάβασης και 2) χρησιμοποιούμε τον ήδη βελτιωμένο κωδικοποιητή όρασης (EV), επίσης παγωμένο, και επανεκπαιδεύουμε το μοντέλο μετάβασης. Όπως αναμενόταν, ο βελτιωμένος κωδικοποιητής όρασης προσφέρει επιπλέον αύξηση στην ακρίβεια ανακατασκευής των προβλέψεων του μοντέλου μετάβασης. Αξίζει να σημειωθεί ότι και οι δύο εκδοχές αποδίδουν καλύτερα από τα συστήματα που έχουν εκπαιδευτεί μόνο σε τυχαίες δράσεις ή ακόμα και μόνο σε ευριστικές δράσεις.

Συμπεράσματα

Στην παρούσα διπλωματική εξετάζουμε εάν ένα ρομπότ μπορεί να αντιληφθεί το περιβάλλον του αποτελεσματικά, αποκλειστικά μέσω της αλληλεπίδρασής του με αυτό, χωρίς να στη-



Σχήμα 10: **Ποιοτική αξιολόγηση:** Η πρωτη και η δεύτερη στήλη δείχνουν εικόνες πριν και μετά την εφαρμογή μιας δράσης. Η τρίτη και η τέταρτη στήλη δείχνουν την πρόβλεψη του μοντέλου μετάβασης όταν αυτό εχει εκπαιδευτεί σε τυχαίες κινήσεις. Στην τέταρτη στήλη φαίνεται η πρόβλεψη του βελτιωμένου μοντέλου. Το βελτιωμένο μοντέλο μετάβασης προβλέπει την κίνηση του αντικειμένου.

ρίζεται σε πρότερη γνώση ή εξωτερική επίβλεψη. Εμπνεόμενοι από την γνωσιακή επιστήμη, η οποία υποστηρίζει ότι η αντίληψη στηρίζεται σε εσωτερικά μοντέλα πρόβλεψης καθώς και στη διαρκή αλληλεπίδραση αισθητηριακών ερεθισμάτων και κινητικής απόκρισης [69, 24], σχεδιάζουμε παρόμοιους μηχανισμούς σε τεχνητούς δράστες και προτείνουμε ένα αυτο-επιβλεπόμενο, αντικειμενο-κεντρικό σύστημα. Το σύστημα αυτό επιτρέπει σε ένα ρομπότ να αναπτύσσεται αυξητικά μέσα από την εσωτερικά παρακινούμενη εξερεύνηση του χώρου του και να βελτιώνει τόσο την δυνατότητα του να ανακατασκευάζει το οπτικό του ερέθισμα, αλλά και να προβλέπει το αποτέλεσμα των δράσεων του.

Στην προσπάθεια αυτή, σημαντική πρόκληση ήταν η απουσία εξωτερικής επίβλεψης. Τα σύγχρονα μοντέλα ΤΝ συνήθως χρειάζονται πολλά και επισημειωμένα δεδομένα για να εκπαιδευτούν κατάλληλα και δυσκολεύονται να γενικεύσουν σε περιπτώσεις όπου είναι διαθέσιμα λίγα δεδομένα, χωρίς επίβλεψη. Για να εκπαιδεύσουμε το μοντέλο όρασης να διαχωρίζει την ει-

κόνα εισόδου σε αντικείμενα χρειάστηκαν πολλαπλές δοκιμές τεχνικών εκπαίδευσης και σφαλμάτων βελτιστοποίησης καθώς και εκτενής βελτιστοποίηση υπερπαραμέτρων, γεγονός που επαληθεύει τα παραπάνω.

΄Αλλη σημαντική πρόκληση ήταν η σχεδίαση ενός σήματος επιβράβευσης που να οδηγεί στην εκπαίδευση εύρωστων πολιτικών. Παρά το γεγονός ότι το σφάλμα πρόβλεψης ως σήμα επιβράβευσης είναι διαισθητικά εύκολα κατανοητό, δεδομένου ότι τα μοντέλα αρχικά είναι αρκετά θορυβώδη το σήμα αυτό δεν αρκεί. Για αυτό, εισάγουμε στο σήμα τους παραπάνω όρους που παρουσιάσαμε νωρίτερα. Παρόλα αυτά, σε ένα πραγματικό περιβάλλον ίσως υπάρχει ανάγκη για μοντέλα με καλύτερες δυνατότητες γενίκευσης ώστε η προτεινόμενη μεθοδολογία να είναι αποτελεσματική. Τελικά, η δουλειά μας από τη μία μεριά αναδεικνύει ότι η μελέτη αρχών από τις γνωσιακές επιστήμες μπορεί να οδηγήσει στη σχεδίαση καλύτερων συστημάτων τεχνητής νοημοσύνης. Από την άλλη, υπογραμμίζει τους περιορισμούς που έχουν τα σύγχρονα μοντέλα ΤΝ και την αδυναμία τους να εκπαιδευτούν τελείως αυτόνομα σε νέα περιβάλλοντα.

Μελλοντικές Επεκτάσεις

Μια φυσική επέκταση της εργασίας είναι η συνεχής μάθηση (continual learning) [70]. Στην συνεχή μάθηση τα μοντέλα ανανεώνονται συνεχώς καθώς νέα δεδομένα γίνονται διαθέσιμα και προσαρμόζονται στις αλλαγές της κατανομής των δεδομένων εκπαίδευσης χωρίς να καταστρέφεται η απόδοση τους στα παλιότερα σύνολα δεδομένων. Στα πλαίσια του πειράματός μας, ο δράστης θα μπορούσε να αλληλεπιδρά με το περιβάλλον του αξιοποιώντας την πολιτική που βασίζεται στα μοντέλα όρασης και επίβλεψης, ενώ παράλληλα τα μοντέλα αυτά εκπαιδεύονται με τα νέα δεδομένα που συλλέγει ο δράστης. ΄Αλλες επεκτάσεις περιλαμβάνουν τη χρήση των αναπαραστάσεων των αντικειμένων για ασκήσεις ελέγχου πάνω σε αυτά τα αντικείμενα και η αξιολόγηση τους με βάση το πως επηρεάζουν τη ταχύτητα εκπαίδευσης των πολιτικών. Η μέθοδος μας επιτρέπει ακόμα μια ποιοτική ανάλυση ενός μοντέλου μετάβασης. Αφού κανείς εκπαίδευσει μια πολιτική με το σφάλμα πρόβλεψης του μοντέλου μετάβασης μπορεί να οπτικοποιήσει τις δράσεις που προτείνει και να κατανόησει καλύτερα τις αδυναμίες του μοντέλου μετάβασης. Τέλος, μία φιλόδοξη εφαρμογή της μεθόδου μας, περιλαμβάνει την ενσωμάτωση της δυνατότητας κίνησης σε ένα ρομπότ το οποίο εξερευνά έναν νέο χώρο. Το ρομπότ θα είναι σε θέση να επιλέξει να κινηθεί ή να αλληλεπίδρασει με τα αντικείμενα που βρίσκονται μπροστά. Τελικά, το ρομπότ θα σχεδιάζει διαδρομές και δράσεις ώστε να εξερευνά καλύτερα το περιβάλλον του, ενώ μαθαίνει ταυτόχρονα και τη δυναμική του.

Προς τη σχεδίαση συστημάτων Τεχνητής Νοημοσύνης επηρεασμένων από τη βιολογία

Η ΤΝ πάντα αξιοποιούσε ιδέες από τη λειτουργία των βιολογικών οργανισμών. Από τις βασικές μονάδες των νευρωνικών δικτύων στις συνελικτικές πράξεις που μιμούνται τον οπτικό φλοιό της γάτας και τους μηχανισμούς προσοχής υπάρχει ένας διάλογος μεταξύ νευροεπιστήμης, συμπεριφορικής ψυχολογίας και ΤΝ. Η ΤΝ αξιοποιεί αυτήν την αλληλεπίδραση για τη σχεδίαση πιο εύρωστων και αξιόπιστων συστημάτων νοημοσύνης. Πιστεύουμε ότι η παρούσα διπλωματική αποτελεί άλλο ένα παράδειγμα αυτής της αλληλεπίδρασης. Αξιοποιώντας αρχές, όπως η

εσωτερική παρακίνηση και η αυτοεπίβλεψη προτείνουμε έναν δρόμο για προσαρμοστικά και αυτόνομα συστήματα ΤΝ, αλλά και ένα τρόπο να αξιολογηθούν θεωρίες για το πως οι οργανισμοί αποκτούν γνώση.

Chapter 1

Introduction

Contents

Today's state-of-the-art AI models continue to break new ground in computer vision and machine learning [71], [3] advancing rapidly across various domains, including image and video classification, semantic segmentation and decision-making. Despite these impressive achievements, the cognitive abilities and world understanding of animals and humans still surpass those of current machine learning (ML) systems. Unlike humans, who require minimal exposure to new tasks to adapt and succeed, ML systems depend on vast amounts of data along with carefully designed supervisory signals from human experts. These data samples must be independent and identically distributed (i. i. d.); when the domain or data distribution shifts typical AI models struggle to generalize effectively. Humans, on the other hand, can master new tasks with limited practice and data. For example, adolescences learn to drive with only about 20 hours of instruction. People can effortlessly navigate in places they have never been before or manipulate objects they have never encountered before, while children learn new tasks fast by re-exploiting structured knowledge from previous interactions [22, 23]. Therefore, it is essential to thoroughly study and draw inspiration from biological cognition and its underlying principles in our attempt to develop more reliable and efficient artificial systems [27].

1.1 Thesis Objectives

In this thesis, we explore active perception through a cognitively inspired approach while leveraging recent advancements in machine learning. Our goal is to emulate the behavior of a human infant, who perceives the world and interacts with it to incrementally enhance their understanding of the environment and develop an internal model of it. Unlike typical AI models that rely heavily on supervisory signals and large datasets, we aim to investigate whether a model can learn in a self-supervised manner within a novel, open-ended environment. To achieve this, we design a training paradigm grounded in well-documented principles from developmental psychology, cognitive science and artificial intelligence.

We aim to leverage several key advantages of physical intelligence, as reported in the literature, to develop the perception capabilities of an artificial system from scratch. First, we will utilize the system's ability to interact with its environment in order to develop embodied intelligence. Next, we will adopt well-established perception principles, enabling the system to perceive the world in an object-centric manner. We will then construct world models that encapsulate the physical properties of the environment, allowing the system to better predict and understand its surroundings. The system will develop incrementally, driven by intrinsic curiosity, which will enable it to naturally refine and enhance its world model over time.

1.2 Cognitive and Developmental Motivations

A key principle supporting our approach is the **embodiment hypothesis**, which suggests that intelligence emerges through an agent's interaction with its environment and as a result of sensorimotor activity [24]. Human infants, for example, are not born with advanced cognitive abilities; instead, they develop incrementally by exploring and interacting with the physical world. This process plays a crucial role in how they eventually develop advanced cognitive skills, often beyond what current AI systems can do. Infants live in a world full of rich regularities that shape their perception, actions and thoughts. Their intelligence is distributed across their experiences and interactions with the environment, thus by enabling an artificial system to intervene in its surroundings, we aim to bootstrap its perception and learning capabilities in a similar manner. Like infants, such a system can benefit from exploration, engaging in diverse, seemingly random and non-goal-directed activities. This behavior promotes open-ended and inventive intelligence development, both in human infants and potentially in artificial intelligence.

Drawing inspiration from physical intelligence, we will also incorporate the concept of **world models** [25, 26]. Humans construct mental models of the world based on sensory information. The brain processes vast amounts of data from daily life, forming abstract representations that guide decisions and actions. Research suggests that perception at any moment is shaped by the

brain's predictions about the future, grounded in these internal models [69]. Animals leverage such world models to quickly acquire new skills, predict the outcomes of their actions and engage in reasoning, planning, exploration and problem-solving. In this thesis, we aim to train world models in an unsupervised manner, use them to promote exploration and iteratively enhance both the system's perception capabilities and the world models themselves [27].

We also adopt an **object-centric approach** to world perception, driven by the dynamics of motion. Objects are central to how humans visually interpret their surroundings [72]. That is why, the question of how humans are able to organize visual input into distinct objects has intrigued researchers for centuries. For instance, the Gestalt school of psychology proposed that humans use cues such as color, texture and motion to group visual elements into coherent objects [73].

Developmental psychology reveals that even in early infancy, humans have an understanding of objects, expecting them to move cohesively along continuous paths, which shapes their perception of object boundaries [72, 74]. Initially, infants may not differentiate between entities apart from their visual boundaries, but through interaction with their environment, they gradually learn to associate specific properties with these entities. For instance, they might discover that spherical objects roll, while objects with rough surfaces are more difficult to push. In this work, rather than viewing segmentation as a passive process, we aim to enable the system to actively interact with its environment to refine its segmentation model. We start from segmenting the visual world into distinct entities and advance toward understanding their physical properties. This progression naturally supports further capabilities, such as using these representations for control tasks and iteratively improving the quality of these representations.

1.3 Overview of our Approach

In this work, we investigate active perception through a cognitively inspired framework that emulates how human infants learn by interacting with their environment. Our goal is to explore whether a model can develop an internal understanding of the world purely through self-supervised interaction, without relying on external supervision or large-scale datasets, in open-ended settings. To achieve this, we study world model learning while a robotic arm actively explores objects on a table so as to maximize an intrinsically motivated epistemic reward function. Importantly, while to our knowledge most previous developmental robotics approaches to world model learning used fixed visual modules [28, 29], here we use the data generated through active interaction with objects to simultaneously learn the world model and the vision module entirely from scratch, without any external supervision, pretrained modules or external datasets. We first show that our proposed epistemic reward generates actions that lead up to three times more object displacement than random actions. We then show that the resulting policy leads to both world model improvement (i.e., better prediction of state-action-state dynamics) and better visual reconstruction.

1.4 Contributions

To validate the proposed methodology, we conduct experiments with a robotic arm in a tabletop environment with a diverse set of objects. We evaluate various versions of our algorithm, exploring different model architectures, training schemes and reward functions. Our results demonstrate that the proposed approach is effective in scenarios where no supervision signals, pretrained models or external datasets are available.

In summary, our key contributions are as follows:

1. **Object-Centric World Modeling:** We adopt an object-centric approach and develop a world model capable of predicting the future states of object representations across frames. Unlike most methods in literature that operate in the high-dimensional visual space, our model reasons in a compact latent space, enabling more efficient and structured dynamics prediction.
2. **Fully Self-Supervised Learning from Scratch:** We train both the vision and world model entirely from scratch, without any supervision or external datasets. Leveraging state-of-the-art self-supervised vision techniques the models are trained entirely on data collected autonomously by the robot through interaction.
3. **Intrinsic Motivation through Predictive Uncertainty:** We design a novel, intrinsically motivated reward signal based on the world model's prediction error, which effectively filters out noise introduced by imperfect models. This reward encourages policies that result in up to three times more object displacement on average.
4. **Data-Driven Model Refinement via Active Exploration:** We show that fine-tuning the models on data collected via the learned policy significantly improves the robot's world understanding. This improvement is quantitatively measured through prediction and reconstruction performance of both the vision and world models.
5. **Comprehensive Experimental Validation:** We validate our approach in a simulated robotic environment, demonstrating the effectiveness of the proposed method.

Chapter 2

Theoretical Background

Contents

In this chapter, we introduce the key concepts and techniques in deep learning that form the foundation of our proposed method. We begin by outlining the different paradigms of machine learning, namely supervised, unsupervised and self-supervised learning, with a particular focus on the latter, as our approach relies exclusively on self-supervised techniques. Next, we describe the primary deep learning architectures employed throughout our experiments, providing the necessary background to understand their role in our framework. We then present DINO, a state-of-the-art self-supervised learning method for representation learning, which we incorporate into some of our vision modules. Finally, we provide a formal introduction to Markov Decision Processes (MDPs) and cover the fundamentals of reinforcement learning (RL), Q-learning and its deep learning-based variant. These will be essential for training the policy that enables our system to interact intelligently with its environment.

2.1 Self-Supervised Learning

Supervised learning is a type of machine learning where models are trained using labeled data. By learning the relationship between inputs and their corresponding labels, a model can generalize to new data and make accurate predictions [30]. In contrast, **unsupervised learning** works without labeled data and aims to find patterns or structures within the data, such as clusters or low-dimensional.

Formally, let $\mathcal{X} \subseteq \mathbb{R}^d$ denote the input space and $\mathcal{Y}$ the output space. $\mathcal{Y}$ is a finite set in case of a classification task. In supervised learning, we are given a dataset of N labeled examples: $\mathcal{D}_s = \{(x_i, y_i)\}_{i=1}^n \subseteq \mathcal{X} \times \mathcal{Y}$ assumed to be drawn i.i.d. from an unknown joint distribution $P(x, y)$. The goal is to learn a function $f : \mathcal{X} \to \mathcal{Y}$ from a hypothesis space $\mathcal{H}$ that minimizes the **expected risk**:

$$\mathcal{R}(f) = \mathbb{E}_{(x,y)\sim P}[\ell(f(x), y)] \tag{2.1}$$

where $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$ is a loss function. Since the true distribution P is unknown, we approximate the expected risk with the **empirical risk**:

$$\hat{\mathcal{R}}(f) = \frac{1}{n} \sum_{i=1}^{n} \ell(f(x_i), y_i) \tag{2.2}$$

The objective is to find:

$$f^* = \arg\min_{f \in \mathcal{H}} \hat{\mathcal{R}}(f) \tag{2.3}$$

Similarly, let $\mathcal{D}_u = \{x_i\}_{i=1}^N \subseteq \mathcal{X}$ represent a dataset of N samples without labels. In unsupervised learning, the goal is to learn some structure or representation from the data without supervision. The objective is to learn a function $f : \mathcal{X} \to \mathcal{Z}$ that maps the input data to some latent space $\mathcal{Z}$, where $\mathcal{Z}$ may represent a lower-dimensional manifold, a set of clusters, or some other learned structure.

In some cases, we seek to minimize the reconstruction loss when the task involves reconstructing the input data from its representation. The objective is then to minimize the expected reconstruction error:

$$\mathcal{R}(f) = \mathbb{E}_{x\sim P(x)}[\ell(f(x), x)] \tag{2.4}$$

where $P(x)$ is the underlying data distribution. Since the true distribution $P(x)$ is unknown, again we approximate the expected loss using the empirical risk:

$$\hat{\mathcal{R}}(f) = \frac{1}{N} \sum_{i=1}^{N} \ell(f(x_i), x_i) \tag{2.5}$$

This empirical risk can take different forms depending on the specific task (e.g., clustering, density estimation, or dimensionality reduction). For example, in clustering, the goal is to learn

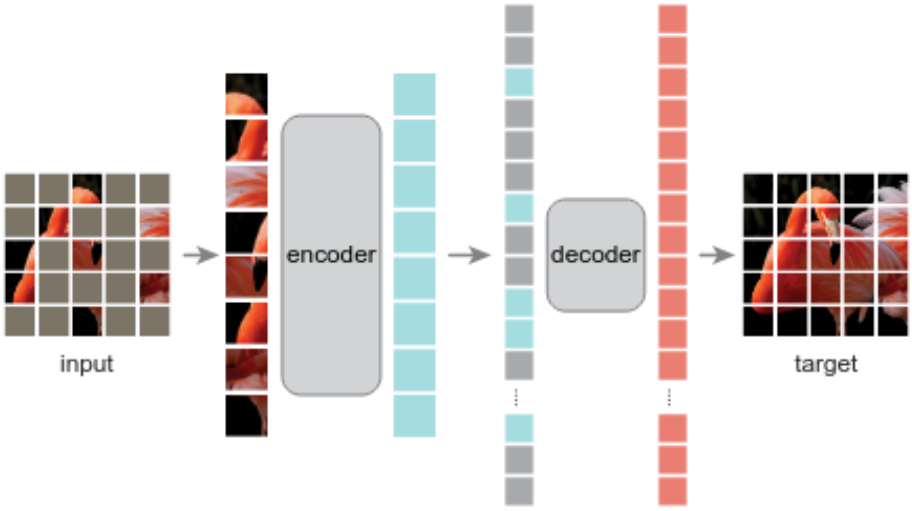


Figure 2.1: A widely adopted SSL architecture [2]: A large portion of the input image is hidden and the network encodes the rest. Then, the decoder tries to reconstruct the original image.

a function f that groups the data into meaningful clusters. This is often done by minimizing a clustering loss, such as:

$$\hat{\mathcal{R}}_{\text{clustering}}(f) = \frac{1}{n}\sum_{i=1}^{n} \ell(f(x_i), c_i) \tag{2.6}$$

where c_i represents the cluster assignment for each sample.

Self-supervised learning (SSL) lies at the intersection of these two paradigms. It uses the structure of unlabeled data to generate pseudo-labels, enabling models to learn tasks that traditionally require supervision. In SSL, supervisory signals are constructed from the data itself, allowing the model to learn meaningful representations without the need for human-annotated labels. In Figure 2.1 a widely adopted SSL framework is presented.

This approach is especially useful in open-ended environments, where labeled data is often unavailable or too costly to collect. In these situations, self-supervised learning (SSL) allows models to learn directly from raw data, making use of the large amounts of unlabeled information and reducing reliance on manual supervision. SSL also resembles how biological agents learn—by recognizing patterns and making predictions from their own experiences, without being explicitly told what to do. In open ended environments, where the task is not fixed ahead of time and the distribution of data is non-stationary or unknown, the traditional supervised and unsupervised learning frameworks fail. In such settings, as in real-world robotics, new situations, objects, or goals can appear without explicit labels. Here, SSL proves particularly powerful: it enables agents

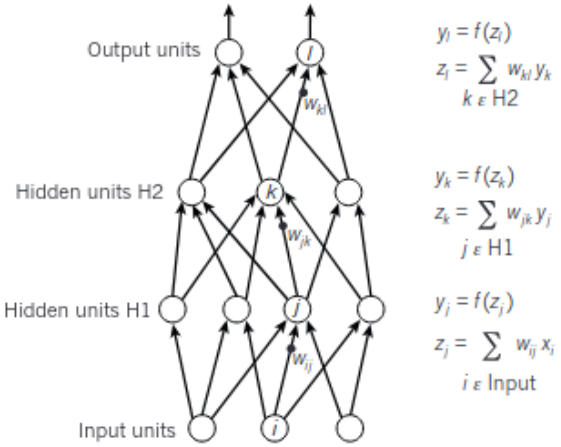


Figure 2.2: The architecture and equations of a two-layered MLP network. [3]

to learn continuously from interaction, treating every experience as a potential learning signal, without requiring manual intervention.

2.2 Common Deep Learning Architectures

Deep learning has achieved remarkable success in recent years across domains such as computer vision, natural language processing and robotics. Key factor behind this success is the development of specialized model architectures that introduce suitable inductive biases for different data modalities and tasks. These architectures are designed not only to approximate complex functions but also to automatically learn rich representations from raw data, significantly reducing the need for handcrafted features [3]. In this section we review the core deep learning architectures used throughout this thesis, focusing on Multilayer Perceptrons (MLPs), Convolutional Neural Networks (CNNs), Vision Transformers (ViTs) and Graph Neural Networks (GNNs).

The simplest deep learning architecture is the **Multilayer Perceptron (MLP)**, Figure 2.2. An MLP is composed of several fully connected layers, where each layer applies a linear transformation followed by a non-linear activation function. The output of each layer serves as the input to the next. Although simplistic, architectures based on MLPs can achieve competitive results in various tasks, such as image classification [31]. However, their fully connected structure results in a large number of parameters and ignores spatial structure in the input data.

Convolutional Neural Networks (CNNs): Convolutional Neural Networks (CNNs) were among the first deep learning architectures to show outstanding performance in computer vision.

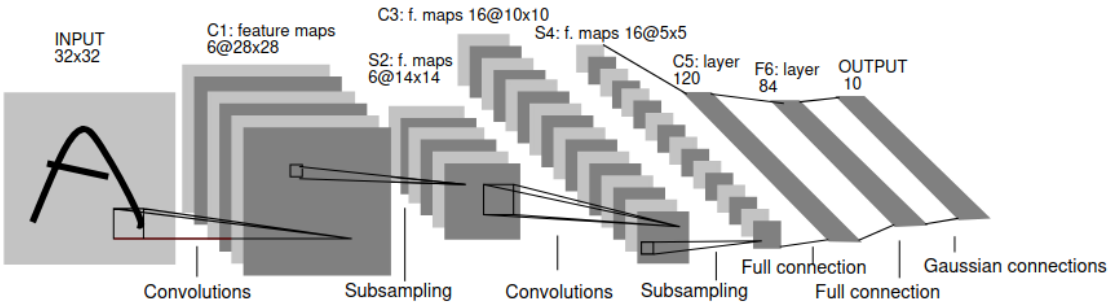


Figure 2.3: Architecture of a widely used CNN: LeNet [4]. In deeper layers the receptive field of each neuron increases.

CNNs are particularly well-suited for processing data with a grid-like topology, such as images [75]. Unlike MLPs, CNNs use convolutional layers instead of fully connected ones, which provides several important advantages:

1. **Sparse Interactions:** In CNNs, connections between input and output units are sparse, achieved through the use of small filters (kernels) that are applied over local regions of the input. This leads to significantly fewer parameters and reduced computational cost, while also preserving spatial locality, allowing the model to capture local patterns effectively.

2. **Translation Equivariance:** Convolutional operations are equivariant to translation, meaning that a shift in the input results in a corresponding shift in the output. This property is especially beneficial for visual tasks, as objects may appear in different positions across images.

3. **Parameter Sharing:** Instead of learning separate weights for every spatial location, CNNs share parameters across the input. A single kernel is used at multiple spatial locations, reducing the number of parameters and allowing the model to generalize learned features across the image. This assumes that a feature useful in one region is likely to be useful elsewhere.

CNNs are also biologically inspired. Their hierarchical structure resembles that of the mammalian visual cortex, where early visual areas detect edges and textures and later areas integrate these into more complex representations like shapes and objects [32]. Similarly, in CNNs, lower layers detect low-level features (e.g., edges), while deeper layers capture high-level semantic concepts.

ResNet, introduced by He et al. [33], is one of the most influential architectures in deep learning and is also employed in this work. ResNet was proposed to address the difficulty of training very deep neural networks, particularly the problem of vanishing gradients [34]. As networks

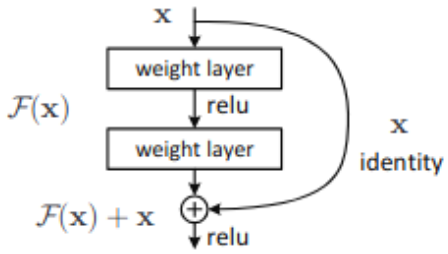


Figure 2.4: Residual building block

grow deeper, gradients propagated during backpropagation can diminish to near-zero, making it difficult to update weights in early layers.

The key innovation in ResNet is the introduction of residual connections, which allow certain layers to be bypassed through identity mappings. These connections create shortcut paths for gradient flow, enabling deeper networks to be trained more effectively. As illustrated in Figure 2.4, the output of a residual block is computed as:

$$y = F(x) + x \tag{2.7}$$

where F represents the composition of a convolutional, a non-linear and a second convolutional function. Residual connections also allow the network to retain and combine both low-level and high-level features. In convolutional networks, the receptive field, namely the region of the input that affects a neuron's activation, grows with depth. By reusing features from earlier layers (with smaller receptive fields), ResNet integrates local detail with global context, enhancing the model's ability to capture complex patterns at multiple spatial scales.

Vision Transformers: While convolutional operations have long been considered essential for image processing, recent advances have challenged this assumption. In particular, Vision Transformers (ViT), introduced by Dosovitskiy et al. [5], demonstrated that transformer architectures [6], originally developed for natural language processing, can also achieve competitive performance on image classification tasks when applied to sequences of image patches. The key idea in ViT is to treat an image as a sequence of patches, analogous to words in a sentence and to process these using the self-attention mechanism. Given an input image $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$, where H and W are the image height and width and C is the number of channels, the image is divided into $N = \frac{HW}{P^2}$ non-overlapping patches of size $P \times P$. Each patch $\mathbf{x}_p^i$ is first flattened from a two-dimensional into a one-dimensional vector and then projected into a D-dimensional embedding $\mathbf{z}^i \in \mathbb{R}^D$ using a learnable linear projection layer E.

To preserve spatial information, a learnable positional embedding $E_{\text{pos}} \in \mathbb{R}^{N \times D}$ is added to the patch embeddings:

$$\mathbf{z}^i = E(\text{flatten}(\mathbf{x}_p^i)) + E_{\text{pos}}(i) \tag{2.8}$$

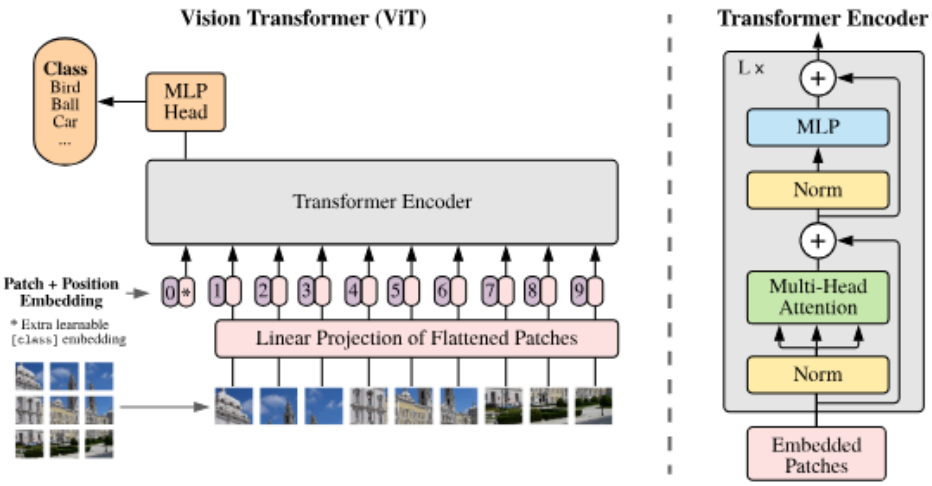


Figure 2.5: The Vision Transformer and Transformer Encoder architectures [5]

These embeddings are then passed through multiple layers of Multi-Head Self-Attention (MHSA) and feedforward networks, forming the core of the transformer architecture. An overview of the Vision Transformer architecture is depicted in Figure 2.5.

Multi-Head Self-Attention: In each attention head h, the patch embedding matrix $Z \in \mathbb{R}^{N \times D}$ is projected into queries Q_h, keys K_h and values V_h using learned projection matrices:

$$Q_h = ZW_{q,h}, \quad K_h = ZW_{k,h}, \quad V_h = ZW_{v,h} \tag{2.9}$$

where $W_{q,h}, W_{k,h}, W_{v,h} \in \mathbb{R}^{D \times D_h}$ and $D_h = \frac{D}{H_a}$ is the dimensionality per head for H_a attention heads.

The attention output for each head is computed as:

$$\text{SA}_h = \text{softmax}\left(\frac{Q_h K_h^\top}{\sqrt{D_h}}\right) V_h \tag{2.10}$$

This mechanism assigns a weight to each token (patch embedding) based on its relevance to others, enabling the model to capture long-range dependencies in the image. In multi-head attention, the outputs of all attention heads are concatenated and projected back to the original embedding space:

$$\text{MHA}(Z) = [\text{SA}_1; \ldots; \text{SA}_H] W_O \tag{2.11}$$

where $W_O \in \mathbb{R}^{(H_a \cdot D_h) \times D}$ is a learnable output projection matrix, as shown in Figure 2.6.

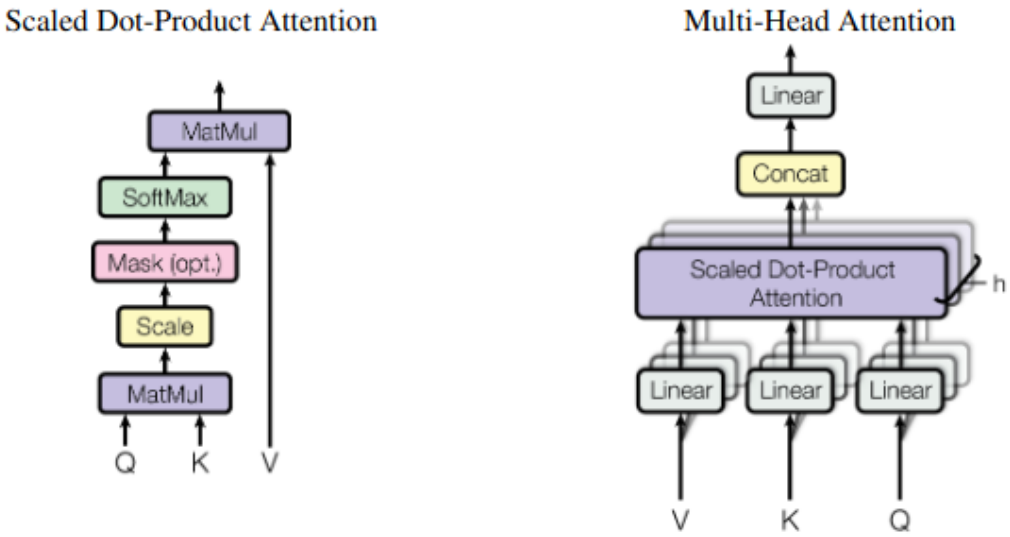


Figure 2.6: Overview of the attention mechanism [6]

The full ViT model includes multiple layers of MHA and feedforward blocks, with layer normalization and residual connections. A special class token is appended to the sequence of patch embeddings to aggregate the global image representation and its final embedding is used for classification.

Graph Neural Networks (GNNs): With the aforementioned architectures one can process Euclidean structured data. However, graph-structured data exhibit irregularities. A graph may contain a variable size of unordered nodes and these nodes may possess different numbers of neighbors. Consequently, operations like convolutions, which are straightforward in the image domain, become challenging when applied to the graph domain. Graphs are ubiquitous. Images, text, chemical molecules or even complex scenes can be treated as graphs. Tasks related to graphs can either have to do with prediction of the attribute and role of a whole graph, a particular node or an edge.

A graph is represented as $G = (V, E)$, where V is the set of vertices (or nodes) and E is the set of edges. Let $v_i \in V$ denote a node and $e_{ij} = (v_i, v_j) \in E$ denote an edge from v_i to v_j. The neighborhood of a node v is defined as: $\mathcal{N}(v) = \{u \in V \mid (v, u) \in E\}$ and the adjacency matrix A is an $n \times n$ matrix, where: $A_{ij} = 1$ if $e_{ij} \in E$ and $A_{ij} = 0$ if $e_{ij} \notin E$.

A fundamental operation in GNNs is that of **message-passing**, shown in Figure 2.7. During each message-passing iteration in a Graph Neural Network, the embedding $h_u^{(k)}$ corresponding to each node $u \in V$ is updated according to information aggregated from u's neighborhood. This

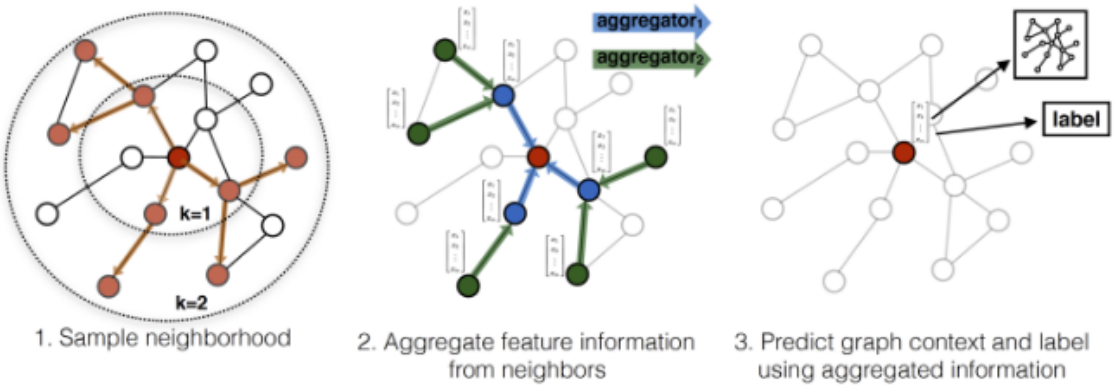


Figure 2.7: The GNN uses message-passing within a node's neighborhood enabling node-level prediction [7].

message-passing update can be expressed as follows:

$$h_u^{(k+1)} = \text{UPDATE}^{(k)}\left(h_u^{(k)}, \text{AGGREGATE}^{(k)}\left(\left\{h_v^{(k)} \mid v \in \mathcal{N}(u)\right\}\right)\right) \tag{2.12}$$

where UPDATE and AGGREGATE are arbitrary differentiable functions (e.g., MLPs).

2.3 Representation Learning with DINO

Combining the strengths of self-supervised learning and Vision Transformers (ViTs), Caron et al. introduced DINO (Self-Distillation with No Labels) [8], a method for training vision models without the need for manual annotations. DINO leverages a self-distillation framework and strong data augmentation strategies to learn rich and transferable image representations.

The core idea behind DINO is to train a student network to match the output of a teacher network, both sharing the same architecture but updated differently. The teacher parameters, θ_t, are updated using an exponential moving average of the student's parameters, θ_{st}, see Eq. (2.13), ensuring stable targets during training. Both networks are fed different augmentations of the same image and the student is trained to produce output distributions similar to the teacher's, as depicted in Figure 2.8.

$$\theta_t = \alpha * \theta_t + (1 - \alpha) * \theta_{st} \tag{2.13}$$

Despite the absence of labels, DINO is able to learn high-level semantic features and object-centric representations, which emerge naturally from the training objective. These features are particularly well-aligned with object boundaries and show strong performance in downstream tasks such as semantic segmentation, image retrieval and classification.

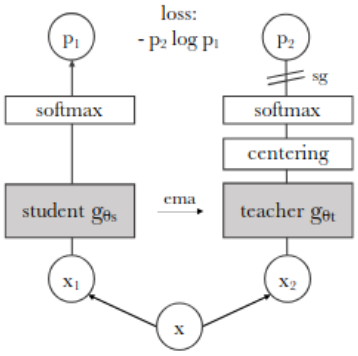


Figure 2.8: Self-distillation with no labels [8]: The input image x is augmented to two different views (x_1, x_2). The student is trained with backpropagation trying to imitate the teacher's output. The teacher is updated based on the student's parameters.

When pretrained on large-scale datasets such as ImageNet [76], DINO-equipped ViTs become powerful feature extractors, offering spatial awareness and generalization capabilities. These characteristics make DINO especially suitable for use in object-centric learning frameworks, such as ours, where identifying and disentangling individual entities in a scene is crucial.

2.4 Reinforcement Learning

Learning through interaction is one of the most intuitive and fundamental forms of learning when thinking about intelligence. An agent observes its environment and interacts with it to discover which actions lead to desirable outcomes. For example, a human learning to drive or engaging in conversation continuously adapts their behavior to optimally influence what happens next.

Reinforcement Learning (RL) is a distinct paradigm of machine learning, alongside supervised and unsupervised learning. It focuses on developing agents that learn to map observations to actions in order to maximize cumulative reward through interaction with an environment. Unlike supervised learning, where labeled input-output pairs are provided, RL does not rely on explicit labels. Moreover, the data in RL is not independently and identically distributed (i.i.d.); instead, it is collected sequentially through the agent's own actions, which influence future states and rewards, making the learning problem significantly more complex [9, 77].

A central challenge in reinforcement learning is the exploration-exploitation trade-off: the agent must decide between choosing actions that are known to yield high rewards (exploitation) and trying out uncertain actions that might lead to better outcomes (exploration). Balancing this

trade-off is essential for discovering optimal behavior, especially in unfamiliar or dynamic environments.

An RL system typically includes the following components:

- Agent and Environment: The agent interacts with an environment by taking actions and receiving observations and rewards in return.
- Policy: A policy defines the agent's behavior; it maps observed states to actions. In Deep Reinforcement Learning (Deep RL), the policy is often parameterized by a deep neural network, which learns to approximate optimal behavior through trial and error.
- Reward Signal: The reward function provides feedback to the agent, guiding learning by defining what constitutes success or failure.
- Value Function: While immediate rewards are important, RL agents must also consider long-term returns. The value function estimates the expected cumulative reward (also called the return) from a given state or state-action pair, helping the agent evaluate the future consequences of its actions.

To formally model the interaction between an agent and its environment in reinforcement learning, we introduce the framework of **Markov Decision Processes (MDPs)**, see Figure 2.9. An MDP is defined by the following components:

- A countable set of states $\mathcal{S}$ (State Space), a subset $\mathcal{T} \subseteq \mathcal{S}$ (known as the set of *Terminal States*) and a countable set of actions $\mathcal{A}$.
- A time-indexed sequence of environment-generated pairs of random states $S_t \in \mathcal{S}$ and random rewards $R_t \in \mathcal{D} \subseteq \mathbb{R}$, alternating with agent-controllable actions $A_t \in \mathcal{A}$, for time steps $t = 0, 1, 2, \ldots$
- The Markov Property, which states that the future state and reward depend only on the current state and action, not on the full history:

$$P\left[(R_{t+1}, S_{t+1}) \mid (S_t, A_t, S_{t-1}, A_{t-1}, \ldots, S_0, A_0)\right] = P\left[(R_{t+1}, S_{t+1}) \mid (S_t, A_t)\right] \quad (2.14)$$

- A termination condition: If an outcome for S_T (for some time step T) is a state in the set $\mathcal{T}$, then this sequence terminates at time step T.

At each time step t, given a representation of the environment's state $S_t \in \mathcal{S}$, the agent selects an action $A_t \in \mathcal{A}(S_t)$. One time step later, the agent receives a reward $R_{t+1} \in \mathcal{R} \subseteq \mathbb{R}$ and transitions in a new state $S_{t+1} \in \mathcal{S}$.

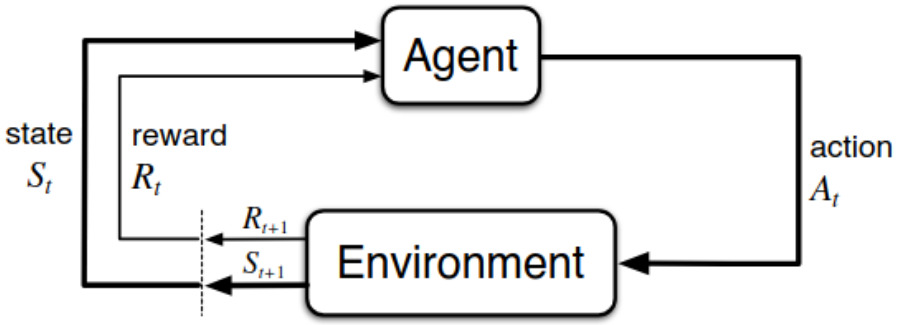


Figure 2.9: The Markov Decision Process framework [9]

Policies and Value functions: A policy $\pi(a|s)$ defines the agent's behavior, specifying the probability of selecting action a in state s. Given a policy, we define the value function $v_\pi(s)$ as the expected return (cumulative reward) when starting from state s and following policy π thereafter:

$$v_\pi(s) = \mathbb{E}_\pi \left[G_t \mid S_t = s \right] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \,\middle|\, S_t = s \right] \tag{2.15}$$

where $\mathbb{E}_\pi$ denotes the expected value under the policy π and $\gamma \in [0, 1)$ the discount factor, controlling how much future rewards are valued relative to immediate ones. Similarly, the $q_\pi(s, a)$ function denotes the expected cumulative reward if an agent is in state s, chooses action a and then follows policy π.

These functions satisfy a fundamental recursive relation known as Bellman equation that is very useful in RL algorithms. It is easy to derive that:

$$\begin{aligned} v_\pi(s) &= \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \,\middle|\, S_t = s \right] \\ &= \sum_a \pi(a \mid s) \sum_{s', r} p(s', r \mid s, a) \left[r + \gamma v_\pi(s') \right] \end{aligned} \tag{2.16}$$

where $p(s', r \mid s, a)$ is the transition-reward distribution: the probability of reaching state s' and receiving reward r after taking action a in state s.

Optimal Value Functions: The optimal value function $v^*(s)$ gives the maximum expected return achievable from state s and is defined as: $v^*(s) = \max_\pi v_\pi(s)$. Likewise, the optimal Q-function $q^*(s, a)$ represents the highest expected return achievable by taking action a in state s and following the optimal policy thereafter: $q^*(s, a) = \max_\pi q_\pi(s, a)$

This optimal Q-function satisfies its own Bellman optimality equation, which is used in many

RL algorithms like Q learning:

$$q^*(s,a) = \mathbb{E}\left[R_{t+1} + \gamma \max_{a'} q^*(S_{t+1}, a') \;\middle|\; S_t = s, A_t = a\right] \tag{2.17}$$

2.5 Deep Q-learning (DQN)

The last equation (2.17) provides an algorithm for iteratively approximating $q^*(s,a)$ by updating an estimation of it, $Q(s,a)$, with a learning rate α:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\left[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)\right] \tag{2.18}$$

Notably, this is an one-step temporal difference (TD) update since the target value is approximated using the current estimate Q function: $v(s_{t+1}) \approx R_{t+1} + \gamma \max_a Q(S_{t+1}, a)$. In Deep Q-Learning, the Q-function is parameterized using a deep neural network, typically referred to as a Q-network, which takes the state s as input and outputs Q-values for all possible actions. While conceptually simple, this direct application of Q-learning with neural networks is unstable in practice due to correlations in sequential data and the moving target problem.

To address these issues, Mnih et al. [35] proposed Deep Q-Networks (DQN) and introduced three key modifications that significantly improved stability and performance, achieving human-level performance on the Atari benchmark:

- A separate, slowly updated target network Q_{target} is used to compute the target value: $R_{t+1} + \gamma \max_a Q_{target}(S_{t+1}, a)$, while the main Q-network is updated via gradient descent. This helps to prevent oscillations caused by rapidly changing targets.
- An experience replay buffer is used to store transition tuples (s, a, r, s') observed during interaction with the environment. During training, batches of transitions are randomly sampled from the buffer, breaking correlations between consecutive samples and reducing the variance of updates.
- Data is collected using an ϵ-greedy policy, where the agent selects a random action with probability ϵ and the greedy action (according to current Q-values) otherwise. This encourages exploration while allowing the agent to learn from off-policy data.

Double Deep Q Learning: A limitation of the original DQN is that the same network is used to select and evaluate the best action in the next state, which can lead to overoptimistic value estimates. To mitigate this, Double DQN [36] was introduced. In this variant, the current Q-network is used to select the best action, while the target network is used to evaluate its value.

The modified target for Double DQN is now:

$$v(s_{t+1}) \approx R_{t+1} + \gamma \max_a Q(S_{t+1}, \text{argmax}_a Q_{target}(S_{t+1}, a)) \tag{2.19}$$

This decoupling of action selection and evaluation leads to more accurate value estimates and improved training stability.

Chapter 3

Literature Review

Contents

This thesis intersects multiple research areas, each of which has been extensively studied in the literature. These fields lie at the intersection of computer vision, robot learning and cognitively inspired machine learning. In the following sections, we provide a comprehensive review of the relevant literature, covering the key problems that this thesis addresses. We begin with unsupervised video object segmentation (3.1), as our problem can be initially formulated as a computer vision task, detecting and segmenting objects in a scene in a self-supervised manner. We then focus on object-centric methods, with particular emphasis on Slot Attention, a prominent approach for learning structured scene representations.

Next, we explore unsupervised reinforcement learning techniques (3.2), which are crucial for enabling our system to acquire and refine its skills in the absence of an explicit reward function. In this process, we leverage world models that operate in an object-aware manner (3.3), using intrinsic rewards (3.4) to guide exploration and learning. Finally, we examine recent deep learning studies where agents learn policies while improving their perception capabilities, which corresponds to an active perception process (3.5). This structured review lays the groundwork for understanding our methodology and its connections to prior research.

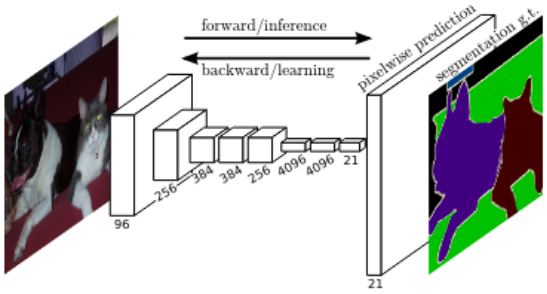


Figure 3.1: A fully convolutional network making segmentation predictions [10]

Figure 3.2: The Davis Dataset typically used for VOS [11]

3.1 Unsupervised Video Object Segmentation

Object segmentation, Figure 3.1, refers to the task of delineating precise object boundaries by classifying every pixel (or unit) in an input image as belonging to a specific object class or the background. In the context of videos, Video Object Segmentation (VOS) aims to extract salient object masks at the pixel level for each frame in a sequence [37]. This task is closely linked to human visual perception; as Biederman [38] suggested, object recognition in images can be conceptualized as the decomposition of regions, often marked by deep concavities, into simple volumetric primitives.

In unsupervised VOS, the challenge is significantly heightened: the model must segment and track salient objects throughout a video without being provided with ground-truth masks or object annotations in the initial frame. This requires the model to not only detect visually prominent entities but also to maintain consistent object identity over time, which is non-trivial in complex or cluttered scenes.

With the rise of deep learning, numerous techniques have been proposed to tackle unsupervised VOS, alongside the development of benchmark datasets for evaluation. Two of the most

widely used datasets are DAVIS (Densely Annotated VIdeo Segmentation) [39] and FBMS (Freiburg-Berkeley Motion Segmentation) [40], both of which provide high-quality, pixel-level annotations across diverse and challenging video sequences. Performance is commonly evaluated using the Jaccard Index (Intersection over Union, IoU), which measures spatial overlap between predicted masks S_{pred} and ground truth S_{gt}, often with consideration for temporal consistency.

$$J(S_{\text{pred}}, S_{\text{gt}}) = \frac{\sum(S_{\text{pred}} \cap S_{\text{gt}})}{\sum(S_{\text{pred}} \cup S_{\text{gt}})} \tag{3.1}$$

One of the earliest unsupervised VOS approaches was proposed by Fragkiadaki et al. [41]. Their method begins by generating object proposals for each frame using either static image boundaries or motion cues derived from optical flow. Moving object proposals are identified using a learnable boundary detector [78] applied to the magnitude of the optical flow field. These initial proposals are refined by a dual-pathway convolutional network operating on both the image and flow fields to assess whether the object is indeed moving. Dense point trajectories are then computed by linking optical flow vectors across time, allowing object proposals to be extended into coherent spatio-temporal tubes.

Temporal coherence in videos presents a powerful cue for object segmentation, as frames, even those that are temporally distant, often exhibit strong structural similarities. Zhou et al. [42] leverage this property through video salient object detection. Their method fine-tunes a pretrained semantic segmentation model to extract spatial saliency features, followed by training a Convolutional LSTM to capture temporal dynamics. While this sequential strategy processes temporal information, it may fall short in modeling long-range dependencies or capturing global consistency across the video.

To address this limitation, Lu et al. [43] propose an attention-based mechanism that models correlations not just between consecutive frames but across multiple frame pairs. By capturing long-range dependencies, their model is able to detect more globally consistent object masks and achieve superior performance compared to earlier methods.

Yang et al. [12] introduce a method that combines optical flow with a transformer-inspired architecture. They frame object segmentation as a flow reconstruction task, allowing the model to differentiate foreground from background. Their method first encodes the input optical flow using a convolutional encoder ϕ_{enc} into features and then introduces two learnable queries that are iteratively updated based on the extracted features. These queries are decoded separately to produce foreground and background segmentation masks. They also propose an additional loss to ensure that the reconstructed optical flow is consistent regardless of the temporal gap it was computed, as shown in Figure 3.3. This approach has demonstrated robust performance across multiple datasets. More recently, Lee et al. [44] proposed a memory-based mechanism to move beyond using only adjacent frame information in unsupervised video object segmentation. Their method extracts feature prototypes from input frames and selectively stores the most informative ones in a memory bank. This memory bank serves as a repository of useful object features from

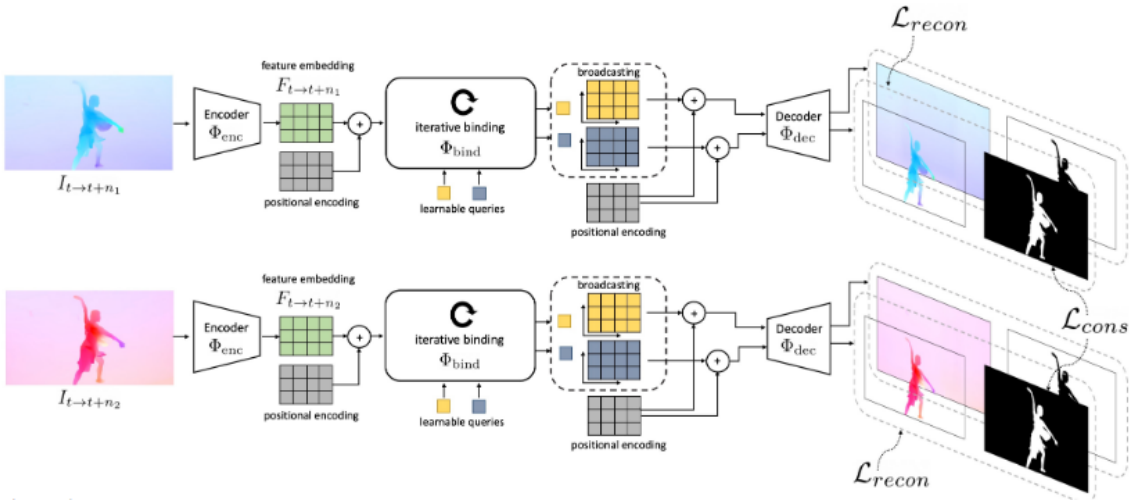


Figure 3.3: Optical flow can be used for efficient foreground-background separation [12].

past frames, enabling the model to leverage longer-term temporal context for improved segmentation. During inference, newly extracted prototypes are combined with stored prototypes and all are ranked based on relevance scores. The top-ranked prototypes are then used to generate the predicted segmentation mask, allowing for more temporally consistent and accurate object tracking across the video.

3.1.1 Object-Centric Unsupervised Object Segmentation

A prominent direction in addressing the unsupervised instance segmentation problem is through object-centric scene representation, where "objects" serve as the fundamental building blocks. Most of these approaches rely on a reconstruction objective to extract meaningful components from visual scenes. The object-centric paradigm aligns with the causal mechanisms that govern our physical world, offering the potential to significantly improve the generalization capabilities of computer vision models [79]. Object-centric scene representation methods can be divided into two main types of models: scene-mixture models and spatial-attention models.

Scene-Mixture Models: Scene mixture models [45, 46, 13] interpret a visual scene as a combination of multiple component images. These components are represented and reconstructed by a generative model that decomposes the scene into distinct elements. One early example of this approach is MoNet [45], a deep generative unsupervised model that uses a variational autoencoder (VAE) [80] paired with a recurrent attention network. MoNet is trained end-to-end without supervision, generating attention masks and reconstructions for image regions. This method is motivated by the advantageous use of compositional structures that break down complex scenes

into simpler, coherent parts with some common structure.

Similarly, IODINE [46] introduces iterative variational inference, where multiple independent latent representations (slots) correspond to different objects in the scene. Each slot generates both a pixel-wise appearance map and a mask assignment for its respective object. However, models like MoNet and IODINE do not account for interactions between scene components. GENESIS [13] addresses this limitation by incorporating a recurrent neural network (an autoregressive prior) that models dependencies between components, enabling the generation of coherent, novel scenes. In Figure 3.4 are shown the corresponding graphical models for each method.

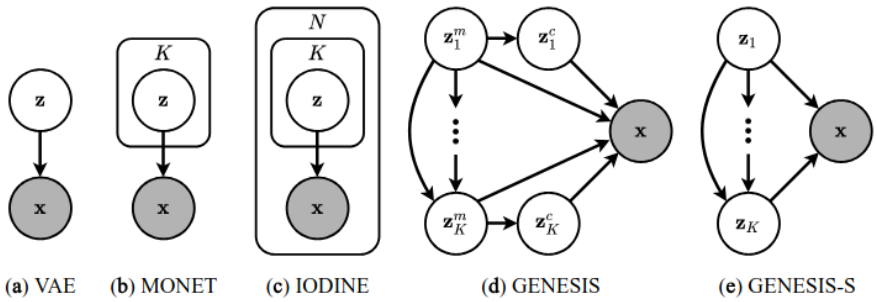


Figure 3.4: Graphical models of different scene mixture models [13]. K denotes the number of the components (slots) in which the scene is decomposed, while N denotes the number of refinement iterations used in IODINE.

While scene-mixture models provide segmentation maps capable of handling objects with complex morphologies, they face several limitations. Each component corresponds to a full-scale image and thus encode object attributes like position and scale only implicitly. Furthermore, scene inference in these models is sequential, which hinders scalability when dealing with scenes containing many objects.

Spatial-Attention Models: In contrast, spatial-attention models [47, 48, 49, 50, 51] directly represent geometric properties of objects, such as position and scale. By grounding object representations in spatial semantics that reflect physical reality, these models promote sample efficiency, interpretability, geometric reasoning, and cross-task transferability. They typically model an image as a composition of background and foreground elements, with the foreground further decomposed into a set of object-centric representations. Each object is described by attributes such as spatial location, visual appearance, and presence.

Visual saliency and multimodal attention, motivated by cognitive and perceptual theories, had already been explored well before the deep learning era. The foundational work by Itti et al. [81] introduced one of the first computational models of saliency-based visual attention, combining

low-level visual cues such as color, orientation, and intensity to predict likely fixations in natural scenes. This model laid the groundwork for numerous saliency algorithms rooted in bottom-up attention. Building on this, works like [82, 83] integrated visual, auditory, and textual streams to model attention and saliency in complex, dynamic environments. These early efforts emphasized the value of attention for video summarization, event detection, and affective analysis, though they relied on hand-crafted features.

Advances in deep learning have enabled more powerful and generalizable attention mechanisms that can be trained end-to-end across modalities and tasks. For example, SUSiNet [84] introduced a unified architecture that jointly learns saliency estimation, action recognition, and video summarization using spatio-temporal features from multiple datasets. Similarly, STAViS [85] proposed a deep audiovisual attention network that combines spatial-temporal visual features with learned auditory saliency to improve localization and attention modeling in videos.

In the context of VOS, SCALOR [48] addresses the scalability challenge by parallelizing both the object discovery and propagation processes, reducing the computational complexity from $O(K)$ in most Scene-Mixture models to $O(1)$, where K is the number of objects in the scene. This parallelization allows the system to efficiently handle scenes with high object density.

However, spatial-attention models often struggle to capture complex objects that cannot be easily represented with simple geometric shapes, such as rectangular bounding boxes. Recent work [51] has demonstrated that this limitation can be mitigated by leveraging object-centric attention maps from self-supervised foundation models, such as DINO, to kickstart unsupervised semantic segmentation. This approach combines the strengths of both object-centric scene representation and large-scale self-supervised learning, pushing the boundaries of unsupervised segmentation capabilities.

3.1.2 Slot Attention Models

A widely-used object-centric method for unsupervised object segmentation and representation learning, which forms the foundation of our approach, is Slot Attention [14]. In this method, a set of variables called slots binds to the perceptual representations of an input scene through a differentiable attention mechanism. These slots serve as object-like entities that capture different parts of the scene. A predefined number K of slots is randomly initialized. Meanwhile, a feature map is extracted from the input image, typically using a convolutional neural network (CNN).

An iterative mechanism is then employed, where each slot attends to features of the input. During this process, slots compete with one another to explain distinct parts of the input scene. They are updated through a gated recurrent unit (GRU) [64], allowing them to progressively refine their representations.

At the end of this procedure a spatial broadcast decoder [65] operates on each slot to reconstruct its corresponding part of the image. The individual reconstructions are then combined into a final, single RGB image, as shown in Figure 3.5.

Figure 3.5: Slot attention [14]: The slots iteratively compete and bind with the representations of the input image. Each slot is then decoded to a specific part of the initial image that corresponds to either an object or the background. The individual reconstructions should be combined into the input image.

DinoSAUR: However, the image reconstruction objective used in Slot Attention models has limitations when applied to real-world data. It is proven successful mainly on simpler datasets, where low-level features like color provide clear cues for assigning pixels to objects. To address this, Seitzer et al. propose an alternative in [15]. Instead of reconstructing the raw input image, their model's decoder reconstructs high-level features derived from self-supervised pretraining. This method, depicted in Figure 3.6 introduces an additional inductive bias by leveraging features that exhibit strong homogeneity within objects, making object segmentation more robust.

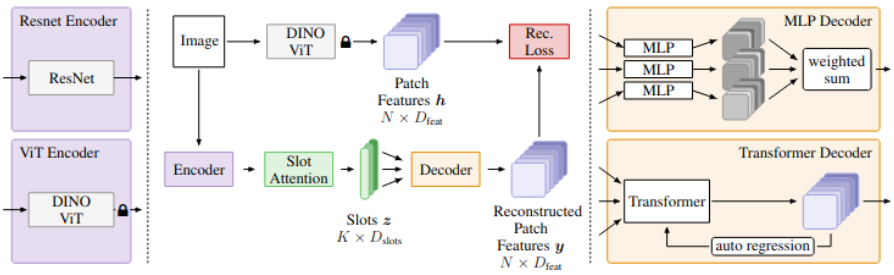


Figure 3.6: Overview of DINOSAUR [15]: DINO extracts object-centric features. Slot attention binds them to slots and is then trained by reconstructing them.

These features can be obtained from modern self-supervised learning techniques, such as DINO [66], which are trained on large datasets like ImageNet. The proposed method can be seen as a form of student-teacher knowledge distillation [86], where the student model (the decoder) compresses the high-dimensional, unstructured information from the teacher's feature space into

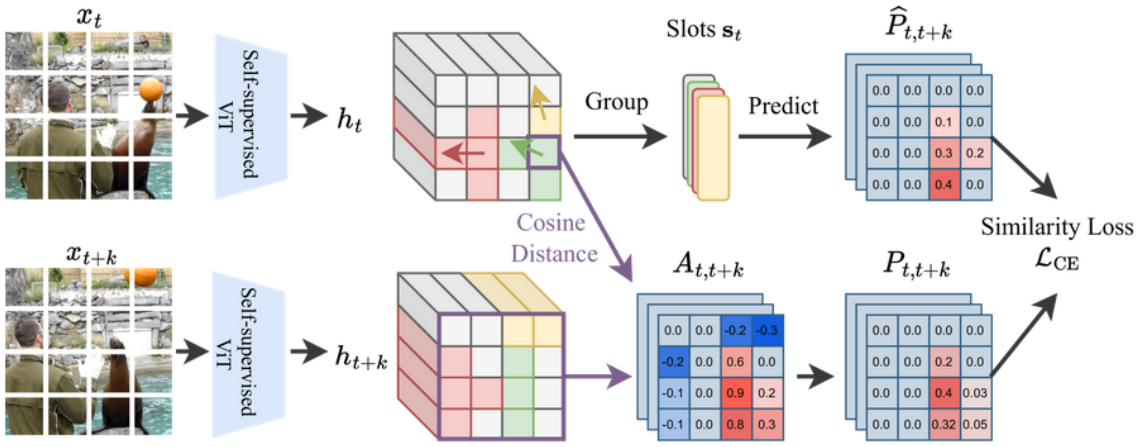


Figure 3.7: Overview of VideoSAUR [16]: Slots should be able to predict a temporal similarity matrix.

a lower-dimensional, structured representation. By doing so, the model gains improved generalization capabilities for complex real-world scenes.

From Images to Videos: A sequential extension of Slot Attention, designed to operate on videos, arises naturally from the need to capture temporal dynamics. In SAVI [67], instead of randomly reinitializing the slots for each consecutive input frame, a predictor module serves as a transition function to model temporal relationships, including interactions between slots. SAVI also incorporates additional contextual information, such as object bounding boxes to condition slot initialization. For training, it uses optical flow predictions as the primary target for each individual frame, encouraging the model to capture temporal consistency. Building on this, VideoSAUR [16] extends DinoSAUR by explicitly integrating temporal information through an additional loss term. This term ensures that the extracted slots can predict an affinity matrix, which encodes the cosine similarity between features from temporally proximal frames. By enforcing this temporal coherence, the model is incentivized to group regions with consistent motion and semantics into the same slots, thus improving object-centric segmentation in video data, see Figure 3.7.

On The Number of Slots: A significant limitation of the object-centric models discussed here, including Slot Attention, is their reliance on a predefined number of slots. This approach requires prior knowledge of the dataset and fails to account for the variability in the number of objects present in each scene. To address this issue, AdaSlot [17] introduces a complexity-aware adaptive slot attention mechanism that dynamically determines the optimal number of slots based on the input data. Initially, a large number of slots is generated, but only a subset is selected for the reconstruction process. The framework also employs a slot sparsity regularization term to balance reconstruction quality with efficient slot usage.

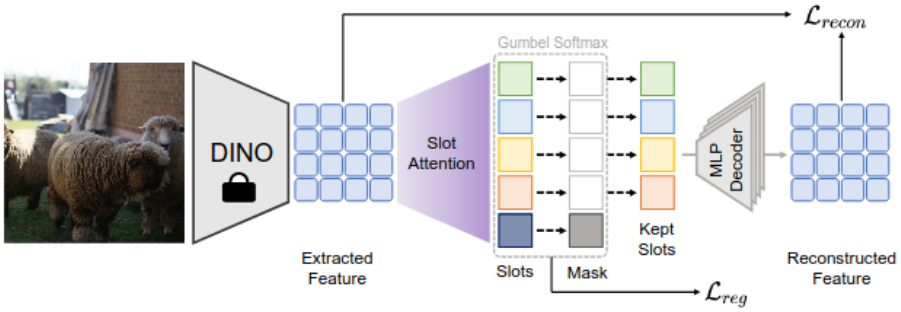


Figure 3.8: Overview of AdaSlot [17]

A lightweight neural network predicts the drop probability for each slot, see Figure 3.8. Applying the Gumbel-Softmax [87] to these probabilities yields a hard but differentiable decision mask. An alternative technique is slot merging, proposed in [88]. This method uses an agglomerative clustering algorithm to merge semantically similar slots based on their cosine similarity. By combining slots that represent the same object, the model can dynamically determine an optimal number of slots, further enhancing segmentation flexibility and efficiency.

As discussed, previous work has addressed the limitations of Slot Attention by incorporating additional information sources, such as motion, depth, or pretraining on large datasets like ImageNet. Similar to color, motion and depth provide useful grouping cues by highlighting objects as they move or stand out in 3D space. However, rather than relying on such auxiliary signals or external datasets, our goal is to develop a system that is fully unsupervised and capable of learning independently.

3.2 Unsupervised Reinforcement Learning of Skills and Representations

The work presented in this thesis aligns with a broader area of research known as Unsupervised Reinforcement Learning. In this paradigm, models are trained to learn meaningful state representations or tasks without access to an externally defined utility signal. Instead, the system relies on intrinsic objectives or self-supervised signals to guide learning. One early example of this approach is Learning to Poke by Poking [52]. In this work, a forward and an inverse model are used to predict the future position of an object being poked. By training a robot to perform poking actions aimed at moving objects to goal locations, the model develops an intuitive understanding of object physics and geometry.

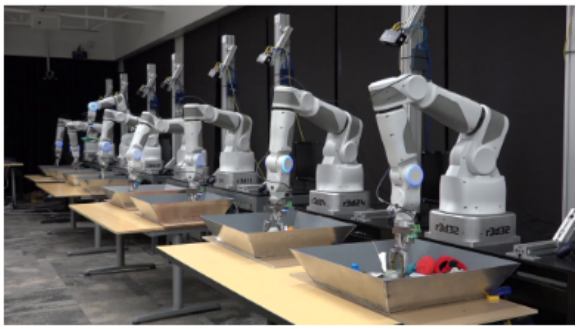


Figure 3.9: Robotic arms interact with their environment and collect data without supervision [18]

In other works [18, 53] off-policy deep reinforcement learning is used to learn closed-loop dynamic visual grasping strategies, using entirely self-supervised data collection as shown in Figure 3.9. These systems generalize to previously unseen objects during testing by predicting the value of low-level end-effector movements directly from raw camera input. Grasping is modeled as a MDP where the state is the perceived image, the actions are Cartesian motions of the gripper and the reward is a self-supervised signal from gripper sensors. The policy is derived using a variation of double Q-learning and stochastic optimization. Building on this paradigm, Grasp2Vec [54] introduces a method to extract object-centric representations from grasping episodes. The goal is to learn embeddings that encode object features, ensuring that images containing the same objects are close together in the embedding space, while images with different objects are far apart. During training, each grasping episode produces an image triplet: $(s_{before}, s_{after}, obs)$. The embeddings of these images represented by ϕ should satisfy: $\phi(obs) \approx \phi(pre) - \phi(after)$. The grasping policy is trained as in [53], with a reward function based on these embedding distances. This approach allows the model to simultaneously learn both a robust grasping strategy and rich object representations.

Object-Centric Unsupervised Reinforcement Learning: Another line of research focuses on learning object-centric representations that are useful for control and reinforcement learning. Representing perceptual observations in terms of entities has been shown to improve data efficiency and transferability across a wide range of tasks. One such approach is SMORL [19], which integrates SCALOR [48], discussed in Section 3.1.1, with goal-conditioned visual reinforcement learning. Observations are encoded as a set of structured vectors z_t through the SCALOR object-centric encoder q_ϕ. These representations are then processed by a goal-conditional attention policy $\pi_\theta(a_t|z_t, z_g)$, where z_g represents the goal state, see Figure 3.10. During training, goal representations are sampled conditionally on the representations of the first observation z_1. At test time, an external goal image o_g is provided, which is processed by q_ϕ to generate a set of potential goal candidates ${z_n}_{n=1}^{N}$. The agent sequentially selects and pursues one goal z_g at a time, effectively breaking down complex tasks into manageable sub-tasks.

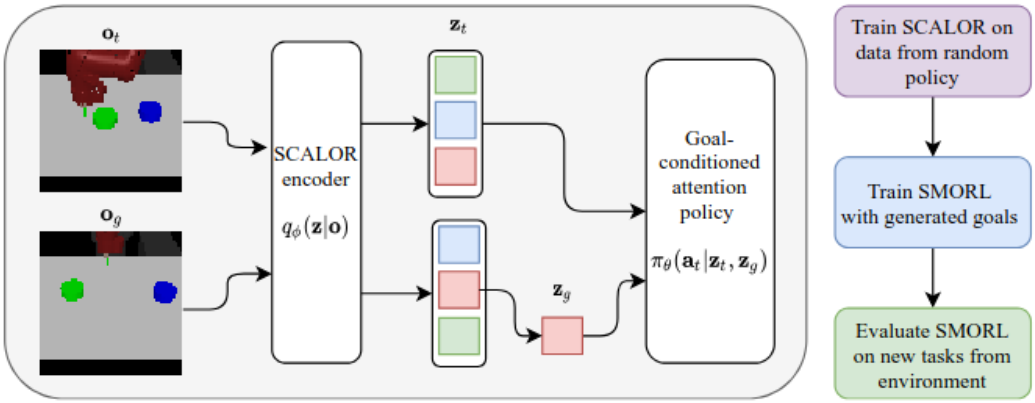


Figure 3.10: The SMORL algorithm enables self-supervised multi-object RL [19]

The object-centric encoder is pretrained on data from random trajectories, while the policy can be optimized using any goal-conditioned model-free reinforcement learning algorithm with randomly sampled goal images. Similarly, Heravi et al. [55] demonstrate the benefits of object-aware representation learning for robotic tasks, particularly in imitation learning. They employ a Slot Attention encoder, similar to ours and show that policies trained on the slot-based representations exhibit improved sample efficiency. They conclude that these policies require fewer demonstration trajectories to learn effectively.

3.3 Object-Centric World Models and Control

Within our system we aim to equip machine learning models with the ability to decompose scenes into objects, infer their properties and understand the relationships between them. Training a world model capable of predicting physical dynamics and the consequences of actions is a key step towards this goal. This world model should be able to answer questions such as, what would happen if an object is pushed in a certain way. Recent research has focused on training such world models operating on object centric state representations, which can be further used for model-based control.

Object-Centric World Models: In Contrastively-trained Structured World Models (C-SWMs) [20], Kipf et al. propose using a graph neural network (GNN) as an action-conditioned transition model. This model learns object-level state abstractions to predict state transitions. The training data consists of offline experience tuples (z_t, a_t, z_{t+1}) where z_t is a state representation, a_t is an action and z_{t+1} is the resulting state after taking action a_t. The transition model T, is implemented as a fully connected Graph Neural Network (GNN), where each node represents an object. The model takes z_t as input and is trained to predict the resulting state z^k_{t+1}. An overview of the

approach is shown in Figure 3.11.

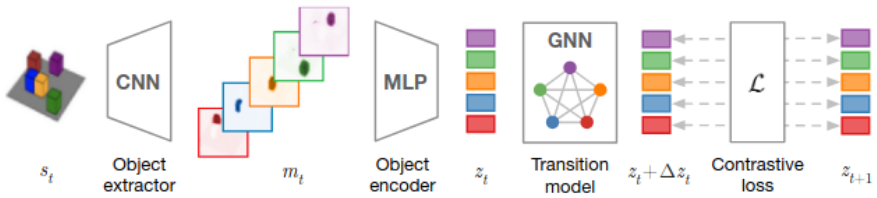


Figure 3.11: The C-SWM model [20]: A CNN and MLP based module extracts and encodes object representations that compose the nodes of a graph. Message passing schemes on that graph model the transition of the objects states.

This GNN framework models pairwise interactions between object while maintaining invariance to the order in which objects are represented. To improve sample efficiency, a contrastive hinge loss is employed. The loss compares the predicted state transition with both the actual next state and a randomly corrupted state representation $\tilde{z}_t$. For a detailed analysis of the graph-based object-centric world model, refer to Section 4.3, where we present our proposed approach.

World Models for Control and Planning: The learned world models can interpolate past experiences and provide analytic gradients of multi-step returns, enabling efficient policy optimization and control. In Dreamer [89], the agent leverages a world model and learns behavior by predicting hypothetical trajectories within the compact latent space of the world model. This approach consists of three key models:

1. A representation model: $p(s_t|s_{t-1}, a_{t-1}, o_t)$, that encodes observations and actions into latent states s_t and an observation model that decodes the state representations back to observations $q(o_t|s_t)$
2. A transition model that predicts future states based on previous states and actions, without observing the actual outcome: $q(s_t|s_{t-1}, a_{t-1})$
3. A reward model that predicts rewards given the current model state: $q(r_t|s_t)$

The observation and reward models are trained in a supervised manner, whereas the transition model is trained so that the predicted state distribution to resemble the state distribution according to the representation model. The objective function is defined as follows:

$$L(\theta) = \mathbb{E}_{q_\theta(s_{1:T}|a_{1:T},o_{1:T})}\Bigg[\sum_{t=1}^{T}\Big(-\ln q_\theta(o_t \mid s_t) - \ln q_\theta(r_t \mid s_t) \\ +\beta\,\mathrm{KL}[q_\theta(s_t \mid s_{t-1}, a_{t-1}, o_t) \,\|\, p_\theta(\hat{s}_t \mid s_{t-1}, a_{t-1})]\Big)\Bigg] \tag{3.2}$$

With this learned world model, the agent can "imagine" the outcomes of potential actions without directly observing them. This capability allows the agent to enhance planning by choosing actions that maximize predicted rewards. Alternatively, the model can generate hypothetical trajectories on which the policy can be trained, significantly improving sample efficiency.

An action model proposes actions that maximize the reward-to-go along these imagined trajectories. The action model is trained by regressing to the values estimated by a critic network. The critic itself is trained to optimize the reward-to-go, which depends on predicted rewards, values and the imagined states and actions. By directly back-propagating gradients through the world model, the critic's objective can be optimized efficiently, further improving sample efficiency.

An alternative approach, depicted in Figure 3.12 is to learn representations and dynamics separately. In Masked World Models [21], a masked auto-encoder (MAE) is used to reconstruct visual observations through convolutional feature masking, while a latent dynamics model is trained on top of the learned visual representations. Decoupling visual representation learning and dynamics learning improves both the quality of representations and model efficiency.

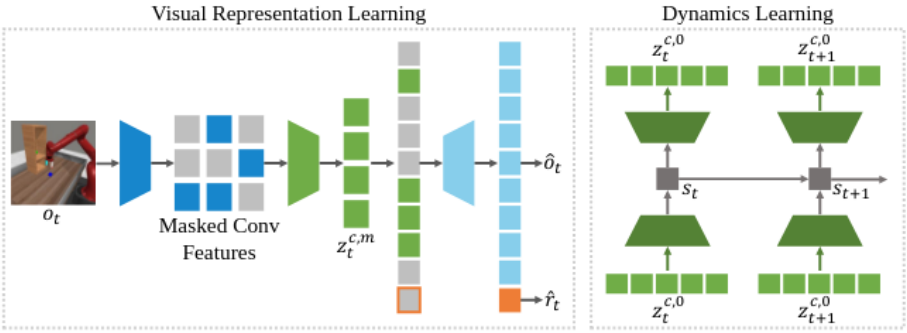


Figure 3.12: Decoupling visual representation learning and dynamics learning improves the quality of representations and model efficiency [21].

Combining the strengths of Slot Attention with advances in Transformer networks, Wu et al. [90] propose SlotFormer, a Transformer-based object-centric dynamics model. SlotFormer models object dynamics as a sequential learning problem, extracting object-centric representations from a sequence of input images and predicting object features at future time steps. By conditioning on multiple frames, the model captures both spatial and temporal relationships between objects, maintaining consistency in object properties and motion across synthesized frames. This capability enables SlotFormer to perform competitively on planning tasks. The aforementioned models leverage object-centric representations and world models in a task specific and reward-dependent way. Instead, in this thesis we propose a task-agnostic, curiosity-uncertainty driven framework to improve a systems understanding of its environment.

3.4 Intrinsically Motivated Reinforcement Learning

Another key component of our approach is the intrinsic reward function, which drives the system's exploration and generates novel, informative trajectories with high information gain for both the vision and world models. In scenarios where extrinsic rewards are sparse or entirely absent, intrinsic motivation becomes crucial, as it encourages the agent to take actions that reduce its uncertainty, improve its ability to predict the consequences of its actions and explore novel states.

Pathak et al. [56, 57] were among the first to effectively integrate curiosity-driven exploration into a reinforcement learning framework. They demonstrated that when sensory inputs are represented in a meaningful way and a forward dynamics model is trained to predict these representations, the model's prediction error can serve as an intrinsic reward signal. Importantly, input representations should filter out information irrelevant to the agent, ensuring that prediction loss is not perturbed by noise in the sensory input. This intrinsic reward function, when combined with a deep reinforcement learning algorithm, enables the agent to explore the state space more effectively and develop policies with greater efficiency.

When the intrinsic reward function relies on prediction error, the agent must perform an action and compute the reward based on the difference between its own prediction and the actual environment behavior. The policy is rewarded if the prediction model and the observed environment disagree, which promotes exploration of unfamiliar or uncertain states. However, this reward depends on the unknown dynamics of the environment and thus the policy requires RL algorithms for training, which typically suffer from sample inefficiency and high-variance.

Later, Pathak et al. [58] address this challenge through a disagreement-based approach. They train an ensemble of world models and incentivize the agent to explore the action space where the models' predictions diverge the most. The intrinsic reward is derived from the variance among the ensemble's predictions, making it fully self-supervised. Additionally, this reward function is differentiable and thus policy optimization can be framed as a supervised learning problem, allowing for more efficient training through direct likelihood maximization. Formally if we denote the ensemble of forward prediction models as $f_\theta = \{f_{\theta_1}, f_{\theta_2}..., f_{\theta_k}\}$ the reward is defined as the variance of the output of the different ensembles:

$$r_t^i = \mathbb{E}_\theta \left[\| f(x_t, a_t; \theta) - \mathbb{E}_\theta \left[f(x_t, a_t; \theta) \right] \|_2^2 \right] \tag{3.3}$$

This reward mechanism drives the agent to explore areas of high uncertainty in the environment, ultimately improving its ability to learn robust and generalizable policies. The policy parameters θ_P can be optimized using direct gradients by treating r_i as a differentiable loss function:

$$\max_{\theta_P} \frac{1}{k} \sum_{i=1}^{k} \left\| f_{\theta_i}(x_t, a_t) - \frac{1}{k} \sum_{j=1}^{k} f_{\theta_j}(x_t, a_t) \right\|_2^2 \quad \text{s.t.} \quad a_t = \pi(x_t; \theta_P) \tag{3.4}$$

Another approach for task-agnostic exploration relies on entropy maximization in the representation space $H(s)$. These methods require a computationally tractable way to estimate the entropy, which can be challenging in high-dimensional spaces. Liu et al. [59] propose a particle-based entropy estimation method, where each state is treated as a particle in the representation space. They approximate the density of the space by calculating the average distance of the k nearest neighbors of a state. For each batch of transitions (s, a, s') sampled from the replay buffer the reward is computed as:

$$r(s, a, s') = \log\left(c + \frac{1}{k} \sum_{z^{(j)} \in N_k(z=f_\theta(s))} \left\| f_\theta(s) - z^{(j)} \right\|\right) \tag{3.5}$$

Here, c is a constant for numerical stability, f_θ is the representation network and and $z^{(j)}$ are the k nearest neighbors (particles) to $f_\theta(s)$ in the batch. Liu et al. demonstrated that this method enables active exploration during unsupervised pretraining, leading to significant improvements in downstream tasks that are otherwise difficult to train from scratch.

3.5 Robotic Active Perception

While most approaches mentioned above study intrinsic motivations to learn policies or world models relying on a fixed perception module, here we are interested in simultaneously improving perception, which corresponds to an active perception process. Several studies in the literature have explored active perception [91, 92, 93]. More recently, researchers have begun integrating deep learning into this concept. Pinto et al [60] in The Curious Robot, argue that biological agents learn visual representations through physical interactions and build a system that pushes, pokes, grasps and observes objects in a tabletop environment to learn such representations. This system is trained to predict the outcomes of these robotic tasks with data annotated in a self-supervised manner. The extracted representations have been shown to be beneficial for simple downstream control tasks.

Similarly, Pathak et al. [61] propose a self-supervised approach to object segmentation through interaction with the environment. The agent here maintains a segmentation hypothesis, manipulates a hypothesized object through random actions and updates the model based on the difference between visual frames captured before and after each action. These interactions provide a noisy yet informative signal that enhances the initial segmentation hypothesis over time.

In [62], Sancaktar et al. demonstrated that a preliminary phase of curiosity-driven free play can enhance downstream task performance. Their system employs an ensemble of world models to plan actions that maximize epistemic uncertainty. The actively collected data is used to iteratively update the models, gradually reducing epistemic uncertainty. Cobra [63] also adopts

a task-free intrinsically motivated exploration approach. Using unsupervised learning, they build object-based transition models of their environment optimizing a pixel-based loss function, which they use in a model-based reinforcement learning setting.

Although similar in spirit, [62] relies on proprioceptive state information to train world models from which rewards are derived, while [63] evaluates their approach only in a two-dimensional, visually simplistic environment. To the best of our knowledge, the work done in this thesis is the first to propose a fully self-supervised, object-centric framework that intrinsically enhances an agent's world perception and modeling in a visually complex environment.

Chapter 4

Methodology

Contents

In this chapter, we present our proposed methodology in detail. We begin with an overview of the full pipeline, followed by a thorough explanation of each component. First, we describe how a vision model can learn to perform object segmentation in a fully self-supervised manner. Next, we introduce a world model trained to predict the dynamics of these object-centric representations. We then explain the motivation and formulation of our intrinsically motivated reward function, along with the reinforcement learning algorithms used to train a policy based on this reward. Finally, we show how this learned policy enables the collection of more informative trajectories, which in turn can be used to further improve both the vision and world models.

4.1 Overview

Our method follows a self-supervised, object-centric pipeline composed of five key stages:

a) We begin by collecting an initial dataset of image sequences, generated by an agent performing random actions in the environment.

b) We train a self-supervised vision model, based on Slot Attention, to segment scenes into object-like components and learn structured object-centric representations.

c) Using the frozen vision model, we train a world model to predict future visual representations conditioned on the agent's actions. This world model is of low quality due to the mainly uninformative actions in the initial dataset.

d) We train a policy using an intrinsic reward derived from the prediction error of the world model, encouraging the agent to explore states where the model is uncertain.

e) Finally, we use the learned policy to collect more informative trajectories involving object interactions and fine-tune both the vision and world model on this richer dataset.

A summary of the proposed framework is illustrated in Figure 4.1.

4.2 Self-Supervised Vision Encoder

To achieve self-supervised, object-centric vision representations, we employ Slot Attention [14], a state-of-the-art unsupervised method for object discovery and segmentation. It introduces latent variables, referred to as slots, which bind to perceptual inputs via a differentiable attention mechanism, capturing distinct parts of the scene as object-like entities.

The core idea is simple: decompose an image into slots and reconstruct it from them. This self-supervised pipeline comprises two main components, a Vision Encoder and a Slot Decoder, enabling Slot Attention to learn object-centric representations without supervision. The Vision Encoder is composed of a deep neural network backbone f_{enc} that extracts feature maps from the input image, which are then processed by the Slot Attention module to produce object-centric representations.

Vision Encoder: Given a video with frames $\mathbf{I}_t$ at timestep t, each frame is encoded into K slots $\mathbf{S}_t \in \mathbb{R}^{K \times D}$, where the number of slots K and the slots' dimensionality D are predefined parameters set by the user. The image is first processed by a DNN backbone, producing a feature map $\mathbf{h}_t = f_{enc}(\mathbf{I}_t) \in \mathbb{R}^{N \times D_f}$, with N spatial locations and feature dimensionality D_f.

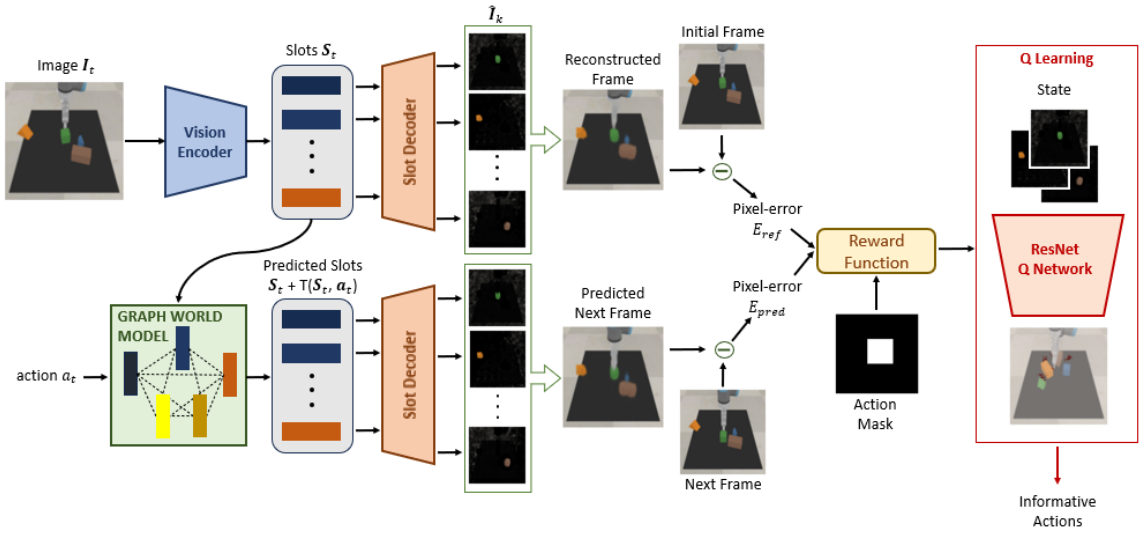


Figure 4.1: **Overview of our proposed framework:** The input image $\mathbf{I}_t$ is processed by the vision encoder to extract K slot representations which can be decoded to the reconstructed image $\hat{\mathbf{I}}_t$. A graph-based world model takes the K slots and the corresponding action as input to predict the future frame $\mathbf{I}_{t+r}$. A reward function is then computed based on the prediction error, which guides a Q-network to propose informative actions.

The Slot Attention module then iteratively binds slots to input features via a differentiable attention mechanism. Slots compete to explain different regions of the scene and are progressively refined through learnable projection layers, an MLP and a Gated Recurrent Unit [64]. This process requires initializing slot representations, originally done randomly.

More specifically, Slot Attention employs an attention mechanism where keys are derived from the feature map $\mathbf{h}_t$ and queries and values come from the current slot representations $\mathbf{S}_t$. This produces attention weights that are used to compute updates for each slot. The formulation is as follows:

$$\mathbf{U}_t = \frac{1}{Z_t} \sum_{n=1}^{N} A_{t,n}\, V(h_{t,n}) \in \mathbb{R}^{K \times D}, \quad \mathbf{A}_t = \text{softmax}_K \left(\frac{1}{\sqrt{D_f}} K(\mathbf{h}_t) \cdot Q(\mathbf{S}_t)^T \right) \in \mathbb{R}^{N \times K} \tag{4.1}$$

Here, K, Q, V are learnable linear projection for keys, queries, values respectively. $\mathbf{A_t}$ represent the attention weights, normalized over slots using a softmax across slot dimension. $Z_t = \sum_{n=1}^{N} A_{t,n}$, is a normalization factor and $\odot$ denotes the Hadamard product. Then the GRU unit updates the slots in an iterative process: $\mathbf{S}_t = GRU(\mathbf{U}_t, \mathbf{S}_t)$. The exact algorithm is described in the pseudocode of Algorithm 1.

Algorithm 1 Slot Attention Module

Input: $\mathbf{h}_t \in \mathbb{R}^{N \times D_f}$, $\mathbf{S}_t^0 \sim \mathcal{N}(\mu, \text{diag}(\sigma)) \in \mathbb{R}^{K \times D}$
Layer Parameters: K, Q, V: linear projections; GRU; MLP; LayerNorm $\times 3$

$\mathbf{h}_t \leftarrow \text{LayerNorm}(\mathbf{h}_t)$
for $i = 1$ to T **do**
  $\mathbf{S}_t^i \leftarrow \text{LayerNorm}(\mathbf{S}_t^{i-1})$
  $attn \leftarrow \text{Softmax}\left(\frac{1}{\sqrt{D}} K(\mathbf{h}_t) \cdot Q(\mathbf{S}_t^i)^\top\right)$ ▷ Normalize over slots
  $updates \leftarrow \text{WeightedMean}(weights = attn + \varepsilon, values = V(\mathbf{h}_t))$
  $\mathbf{S}_t^i \leftarrow \text{GRU}(\text{state} = \mathbf{S}_t^{i-1}, inputs = updates)$
  $\mathbf{S}_t^i \leftarrow \mathbf{S}_t^i + \text{MLP}(\text{LayerNorm}(\mathbf{S}_t^i))$
end for
return $\mathbf{S}_t^T$

Slot Decoder: Each slot is decoded using a spatial broadcast decoder [65] to reconstruct its corresponding scene region. Each slot $\mathbf{s}_k$ produces a reconstruction $\hat{\mathbf{I}}_k$ and a mask $\mathbf{\Pi}_k$, normalized via spatial softmax. The final image is then formed by combining all components:

$$\hat{\mathbf{I}}_k, \hat{\mathbf{\Pi}}_k = f_{dec}(\mathbf{s}_k), \quad \mathbf{\Pi}_k = \text{softmax}_K \hat{\mathbf{\Pi}}_k \quad \hat{\mathbf{I}} = \sum_{k=1}^{K} \mathbf{\Pi}_k \odot \hat{\mathbf{I}}_k. \tag{4.2}$$

For simplicity, we use $\hat{\mathbf{I}}_t = f_{dec}(\mathbf{S}_t)$, where f_{dec} denotes the entire slot decoder pipeline.

Training through Reconstruction: We can now train the CNN backbone of the encoder, the trainable parts of the slot attention, as well as the decoder, jointly by simply reconstructing the original input, across all frames:

$$L_{rec} = \sum_{t=1}^{T} \|\hat{\mathbf{I}}_t - \mathbf{I}_t\|^2 \tag{4.3}$$

Modifications on Slot Attention: While object-centric learning shows promise, the field remains relatively immature as most methods rely on weak supervision, such as optical flow or depth maps, or large-scale pretrained encoders. We instead train Slot Attention **from scratch** using only data collected autonomously by a robotic arm.

To improve the performance and stability of our learning framework, we introduce two key modifications to the standard pipeline:

1. **ResNet Encoder:** Rather than using a Vision Transformer (ViT) backbone, commonly employed in conjunction with DINO pretraining on large-scale datasets like ImageNet, we opt

for a ResNet-based encoder trained directly on our task-specific data. This choice is motivated by several factors. First, convolutional networks like ResNet inherently incorporate strong spatial inductive biases that are beneficial in low-data regimes, unlike transformers which typically require extensive pretraining to generalize effectively. Second, the hierarchical feature extraction and large receptive fields of ResNet layers help capture object-level context in cluttered scenes, making them well-suited for structured visual inputs such as tabletop environments with multiple interacting objects. Empirically, we find that the ResNet encoder not only performs more reliably than the ViT alternative in our setting but also leads to improved reconstruction and segmentation outcomes.

2. **Self-Supervised Pretraining via Autoencoding:** To further enhance the representational capacity of the encoder and promote stable convergence during Slot Attention training, we introduce a self-supervised pretraining phase. Specifically, we use an autoencoding setup where the ResNet encoder is paired with a symmetric ResNet-style decoder composed of deconvolutional layers. This architecture is trained to reconstruct the input images using only reconstruction loss, without any labels or object-specific annotations. Through this unsupervised pretraining, the encoder learns to capture general-purpose visual patterns that serve as a strong initialization for downstream object-centric learning. This pretraining phase not only accelerates convergence but also mitigates common failure modes such as slot collapse and noisy segmentation in the early stages of Slot Attention training.

Entropy-based loss term: Optimizing Slot Attention to produce clean, localized masks remains challenging in our setting, thus we introduce an additional loss term that penalizes the entropy of the spatial masks $\mathbf{\Pi}_k$ associated with each slot. This promotes low-entropy, coherent masks, reducing noise and improving segmentation. The proposed loss term for pixel (i, j) is formulated as $L_{ent}^{ij} = -\sum_{k=1}^{K} \mathbf{\Pi}_k * log(\mathbf{\Pi_k})$. The final objective is now:

$$L = L_{rec} + \beta * \sum_{i,j} L_{ent}^{ij} \tag{4.4}$$

where b is a hyperparameter we set to 10^{-3}.

From Images to Videos: To capture temporal dynamics, we employ a sequential extension of Slot Attention, designed to operate on videos. Instead of randomly re-initializing the slots for each consecutive input frame, a predictor module serves as a transition function to model temporal relationships, as done in SAVI [67]. The slots for frame $t + 1$ are initialized using a transformer-based mechanism as follows:

$$\tilde{\mathbf{S}}_t = \text{LN}\left(\text{MHSA}(\mathbf{S}_t) + \mathbf{S}_t\right) \tag{4.5}$$

$$\mathbf{S_{t+1}} = \text{LN}\left(\text{MLP}(\tilde{\mathbf{S}}_t) + \tilde{\mathbf{S}}_t\right) \tag{4.6}$$

Here, MLP denotes the Multilayer Perceptron, MHSA is the multi-head dot-product self-attention mechanism [6] and LN is layer normalization [94] applied after each residual connection.

4.3 World Model for Predicting Future Slots

Our goal is to enable models to decompose scenes into objects, infer their properties and understand inter-object relations. A key step is training a world model that predicts physical dynamics and action outcomes—e.g., what happens when an object is pushed. Given that our system already extracts structured slot representations, building such a model becomes straightforward.

Following Kipf et al. [20], we use a fully connected graph neural network (GNN) as an action-conditioned transition model over slot representations. It learns object-level abstractions from offline tuples $(\mathbf{S}_t, a_t, \mathbf{S}_{t+r})$, where $\mathbf{S}_t$ are the slots at time t, a_t is the action and $\mathbf{S}_{t+r}$ the resulting slots after a fixed interval r, corresponding to the frame where the action's effect is observed. The model predicts transitions such that $\mathbf{S}_t + T(\mathbf{S}_t, a_t) \approx \mathbf{S}_{t+r}$.

Implementation-wise, a node update function f_{node} and an edge update function f_{edge} is shared across all nodes and edges, both implemented as MLPs. A single round of message passing updates is performed using the following equations on individual slots $\mathbf{s}_t^k$: $e_t(i,j) = f_{edge}([\mathbf{s}_t^i, \mathbf{s}_t^j])$ and $\Delta\mathbf{s}_t^j = f_{\text{node}}([\mathbf{s}_t^j, a_t^j, \sum_{i \neq j} e_t(i,j)])$.

Typically, we can train the world model using the MSE loss over the predicted slots:

$$L_{pred} = \|\mathbf{S}_t + T(\mathbf{S}_t, a_t) - \mathbf{S}_{t+r}\|_2 \tag{4.7}$$

To improve sample efficiency, a contrastive hinge loss is also employed, where the predicted state transition is compared to a randomly corrupted state representation $\mathbf{S}^c$:

$$L_{hinge} = max(0, \gamma - \|\mathbf{S}_t + T(\mathbf{S}_t, a_t) - \mathbf{S}^c\|_2) \tag{4.8}$$

We also introduce a third reconstruction loss term back on the pixel space employing the frozen vision decoder f_{dec} and ensuring that the predicted slots can be decoded to actual changes in the robot's environment:

$$L_{rec} = \|f_{dec}(\mathbf{S}_t + T(\mathbf{S}_t, a_t)) - \mathbf{I}_{t+r}\|_2 \tag{4.9}$$

Overall, the loss is defined as $L_{wm} = L_{pred} + L_{hinge} + \alpha L_{rec}$. In practice, we used $\gamma = 10$ and $\alpha = 10^3$.

The standard prediction loss L_{pred} encourages the world model to accurately estimate how object slots evolve over time under the influence of actions. However, minimizing only this loss can lead to degenerate solutions, especially in noisy or low-signal environments where the model might overfit to trivial dynamics (e.g., ego-motion). The contrastive hinge loss L_{hinge} explicitly pushes the predicted future slots away from randomly corrupted states encouraging the model to learn more discriminative and robust object-level dynamics. Additionally, the reconstruction loss L_{rec} in the pixel space, ensures that latent predictions correspond to visually meaningful outcomes when decoded, aligning the latent space dynamics with observable changes in the environment.

Together, these auxiliary losses help regularize training and promote the emergence of physically coherent object dynamics.

4.4 Designing an Intrinsically Motivated Reward

A key component of our approach is an intrinsic reward function that guides exploration by encouraging trajectories with high information gain for both the vision and world models. We examine a variety of rewards that reflect the uncertainty of our models, such as (i) the reconstruction error of the vision encoder L_{rec}, (ii) the prediction error of the world model and (iii) the prediction variance of an ensemble of world models.

The underlying intuition is that high reconstruction or prediction error signals unfamiliar or poorly understood regions of the state space. These regions are likely to provide informative learning signals when explored.

1. **Vision Model Reconstruction Error L_{rec}:** When the vision encoder reconstructs an input frame with high error, it implies that the observed scene is novel or significantly different from previously encountered ones. Using this reconstruction loss as a reward signal encourages the policy to visit diverse visual states, thereby facilitating the learning of a more comprehensive and generalizable representation.
2. **World Model Prediction Error:** Similarly, large prediction errors from the world model suggest that the environment responded to the agent's actions in an unexpected way. This indicates that the action led to novel outcomes, which can improve the model's understanding of object dynamics. We therefore treat high prediction error as an intrinsic reward signal that encourages exploratory behavior.
3. **Disagreement in an Ensemble of World Models:** Inspired by prior work on uncertainty-driven exploration [58], we train an ensemble of three world models with identical architecture and objectives but initialized with different random seeds. The variance or disagreement among the predictions of the ensemble serves as a proxy for epistemic uncertainty. Actions leading to high prediction disagreement are considered more uncertain and are thus rewarded more. This approach encourages the agent to seek states where the model is less confident, leading to richer and more informative data collection.

We analyze and compare the effectiveness of these different intrinsic reward signals in guiding exploration in Section 5.4, highlighting their respective impacts on policy behavior and model learning.

The proposed reward: Building on our statistical and empirical analysis, we base the final proposed reward on the world model's prediction error. Raw prediction error can be noisy—especially

early in training—due to limitations in the models. To address this, we compute the error in pixel space as the difference between predicted and actual future frames:

$$\mathbf{E}_{pred} = (f_{dec}(\mathbf{S}_t + T(\mathbf{S}_t, a_t)) - \mathbf{I}_{t+r})^2 \tag{4.10}$$

However, this error may also capture biases introduced by the decoder, reflecting the limitations of the current vision model rather than true prediction error. To isolate this, we compute a reference error using only the static vision reconstruction pipeline:

$$\mathbf{E}_{ref} = (f_{dec}(\mathbf{S}_t) - \mathbf{I}_t)^2 \tag{4.11}$$

Only prediction errors exceeding this reference reflect epistemic uncertainty, namely the model's current ignorance about the outcome of its own actions.

To localize the learning signal, the reward is computed only over a region of the image centered on the action, denoted by a binary spatial mask $\mathbf{M}$. This reduces the influence of irrelevant changes and focuses the reward on action-relevant areas. Let p denote the pixel location corresponding to the center of the robot's action in the image space. The spatial mask $\mathbf{M}_{p,d}$ is defined as: $\mathbf{M}_{p,d}(x) = \mathbf{1}_{\|x-p\|_1<d}$, where d is a predefined radius and $\|\cdot\|_1$ denotes the L1 (Manhattan) distance. The mask assigns a value of 1 to all pixels within distance d of the action point and 0 elsewhere. Overall, the intrinsic reward is calculated as:

$$r_t = \sum_i \sum_j [\mathbf{M} \odot max((\mathbf{E}_{pred} - \mathbf{E}_{ref}), 0)]_{ij} \tag{4.12}$$

A deeper intuition behind the designing of the reward function is given in Section 5.4.

4.5 Training a Policy

In order to train a policy to maximize the proposed reward, we frame the problem as a Markov Decision Process (MDP). States are defined by the $K = 10$ segmentation masks, concatenated in the channel dimension, which are produced by the frozen vision model at each timestep and actions correspond to the robotic arm's movements. We train the policy using Double Q-learning [36], with the Q-network implemented as a ResNet. The network takes as input the segmentation masks and outputs a spatial map indicating the expected value of performing a pushing action at each pixel.

More specifically, the state input during policy learning is defined as $s_t = \{\Pi_k\}$, where $\{\Pi_k\}$ are the K segmentation masks produced by the vision model at time t. In principle, other representations could also be used as state input, such as the raw image $\mathbf{I}_t$, the image features $\mathbf{h}_t$, or

the slot embeddings $\mathbf{s}_k$. However, we choose to use the segmentation masks because they provide a spatially structured and object-centric representation, which simplifies the action selection task. The reward signal is computed intrinsically, as described in previous sections, and the actions correspond to physical interactions between the robot and its environment.

The action space consists of 2D spatial locations corresponding to pixels in the image plane. Each action represents a pushing motion executed by the robot at a specific location. The Q-network outputs a dense spatial map of Q-values, indicating the expected return of performing a push at each pixel. During execution, one of the pixels with the highest Q-value is selected randomly and the corresponding action is translated into a physical robot movement using a predefined mapping from image coordinates to the workspace of the robotic arm, see Section 5.1.

The Q function $Q : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ calculates the quality of a state-action combination. The expected reward for state-action is computed as $\tilde{r}_t = r_t + \gamma Q_{\theta^-}(s_{t+1}, \text{argmax}_a Q(s_t, a))$, where θ^- is a previous version of the ResNet parameters. The Q-function is updated as described in Section 2.5, with α_t denoting the learning rate:

$$Q_\theta(s_t, a_t) = Q_\theta(s_t, a_t) + \alpha_t \left(\tilde{r}_t - Q_\theta(s_t, a_t)\right) \tag{4.13}$$

4.6 Refining the Models with Informative Trajectories

The learned policy generates image sequences that are novel and informative for both the vision and world model. Using the frozen policy the agent interacts with its environment. Due to the way the reward function was designed, the agent now performs actions that often result in environmental changes the world model cannot accurately predict. Consequently, the collected trajectories not only lead to broader exploration of the environment but also systematically expose the limitations of the current models.

We exploit these informative interactions by fine-tuning both the vision and world models on the new dataset. Unlike the initial dataset, which was largely composed of static frames or simple gripper motion, the new trajectories feature rich, object-centric interactions. This increased diversity allows the models to observe dynamic object behavior and previously unseen spatial configurations. Specifically, in our setting these more informative trajectories correspond to robot actions that with high probability cause the movement of the objects in the robot's environment with respect to random actions that most of the time do not interact with an object.

We exploit these informative interactions by fine-tuning both the vision and world models on the new dataset. Unlike the initial dataset, which was largely composed of static frames or simple gripper motion, the new trajectories feature rich, object-centric interactions. This increased diversity enables the models to witness dynamic object behavior and encounter spatial configurations that were previously absent during training. In our setup, these more informative trajectories

arise from robot actions that are far more likely to cause actual object displacement, unlike random actions, which often fail to make meaningful contact with objects.

As a result, both models experience measurable improvements in performance:

- The vision model enhances its ability to reconstruct complex scenes.
- The world model becomes better at capturing object dynamics and predicting future states.

Model uncertainty is used to guide exploration. Then, data collected in this exploratory process are used for a targeted refinement of the models. In this way, we establish a self-improving learning loop that builds world understanding without any external supervision.

Chapter 5

Experiments

Contents

To evaluate our proposed methods and hypotheses, we designed an appropriate experimental setup using the CoppeliaSim simulation environment [68]. The experiment involves a UR5 robotic arm, Figure 5.1, interacting with objects placed on a tabletop. These objects are simple three-dimensional geometric shapes with unknown colors and physical properties. The robotic arm executes non-preemptive motion primitives, parameterized by three values: the x and y coordinates on the table and the movement orientation. The orientation is selected from 16 discretized options. The only type of action performed is pushing. To simplify the experimental setup, we restrict our experiments to pushing actions with a fixed orientation.

The primary objective of our system is to detect objects in the scene and infer as much as possible about their physical properties. A key challenge in this problem setting is that our models have no access to any supervision signal or prior knowledge. *Instead, all models are trained from scratch, relying entirely on the sequences of interactions experienced by the robotic arm.*

Initially, the robotic arm interacts randomly with the environment, selecting from the available

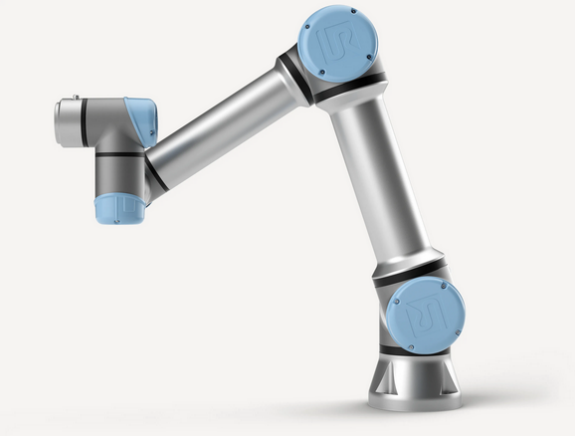


Figure 5.1: The UR5 robotic arm. We equip the robot with a gripper.

pushing actions. During this phase, we construct a dataset comprising tuples of robotic actions and corresponding image sequences. This dataset, which we call *initial dataset*, is then used to train our vision and world models. We also construct a control (oracle) dataset, in which the robotic arm either interacts randomly with the environment or pushes the objects in front of it using heuristics. We demonstrate a few instances in Figure 5.2.

5.1 Collecting Trajectories in the Simulation Environment

We conduct all experiments in a simulated tabletop environment built using CoppeliaSim, a widely adopted robotic simulation platform. CoppeliaSim combines Python and Lua scripting with C/C++ plugins and utilizes the MuJoCo physics engine [95] for accurate rigid body dynamics, as well as dedicated engines for forward and inverse kinematics computations. The environment supports multi-threaded execution, allowing robotic interaction to occur concurrently with model training and data processing, which enables efficient data collection and learning cycles. Our environment setup and control logic are based on the open-source repository provided in [96].

Motion Primitives: To enable the robot to interact with objects, we define pushing actions as parameterized motion primitives. Each action is defined by a target pixel location and an orientation, specifying the direction of the push. Rather than planning actions online, we use a predefined motion sequence executed by the UR5 robotic arm with a closed gripper. The length and trajectory of the push action are fixed.

To execute the action, the selected pixel point is transformed into a 3D coordinate in the robot's

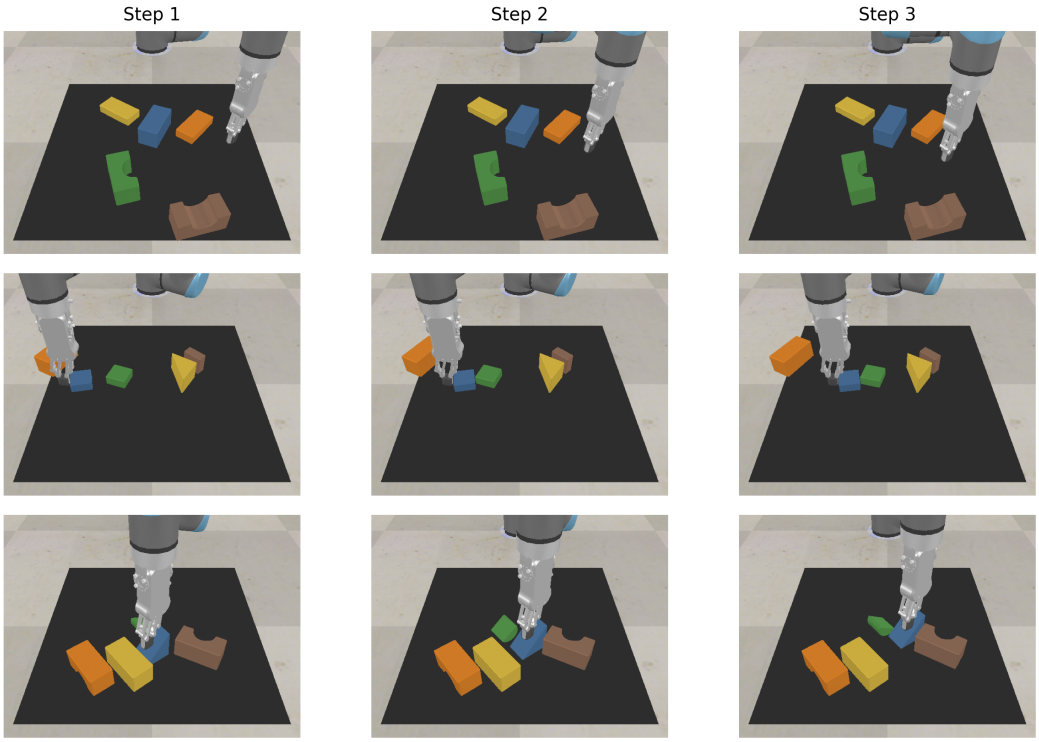
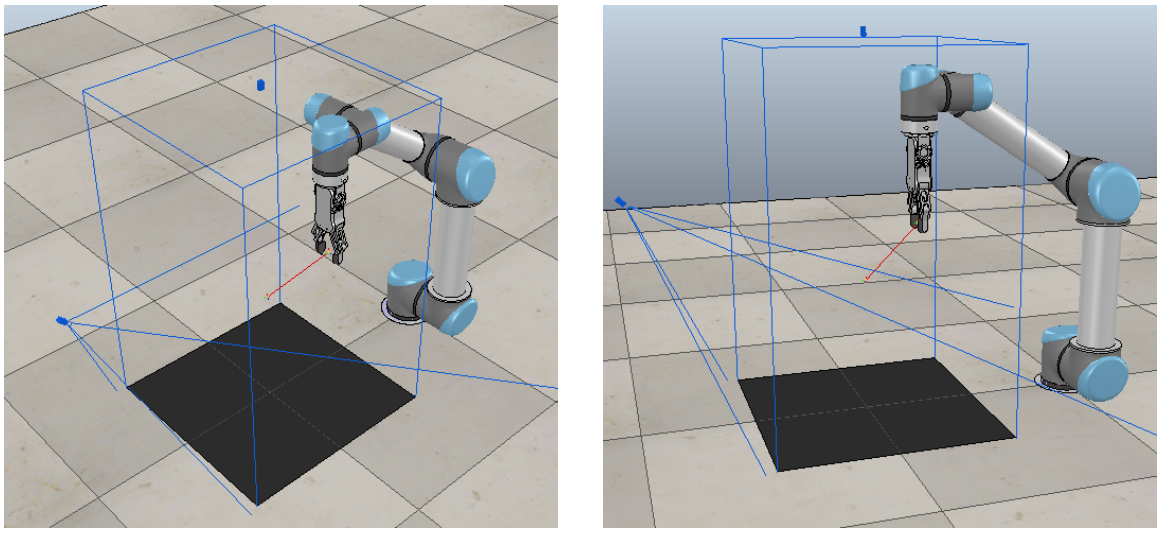


Figure 5.2: **Experimental Setup:** Sequences of robotic arm push actions in our tabletop environment. Top: random push with no object movement. Middle and bottom: heuristic pushes causing object movement.

reference frame using the known workspace limits. The height of the push i.e., the vertical distance from the tabletop, is determined using a heightmap obtained from a simulated depth sensor. This ensures that the pushing motion occurs just above the object surface and is physical plausible.

Camera Setup and Coordinate Transformations: The simulation environment is equipped with a four-channel RGBD camera sensor mounted above the scene. To enable interactions based on visual input, we perform coordinate transformations between the robot's workspace and the camera view.

Specifically, to execute a pushing action at a pixel location selected in the camera frame, we transform this point into the robot's coordinate frame. Conversely, when interpreting the results of robot actions or computing rewards, it is often necessary to map a 3D location in the robot's reference frame back to a pixel location in the image. These forward and inverse transformations are computed using the intrinsic parameters of the camera, such as focal length and principal point, along with known extrinsic calibration between the camera and the robot base.



(a) View 1 (b) View 2

Figure 5.3: The CoppeliaSim simulation environment: Our scene consists of the UR5 robotic arm and two camera sensors. Coordinate transformations from robot's workspace to camera view are needed.

Projection of Image Points into the Robot's Coordinate Frame: We want to perform a geometric transformation of a point from 2D pixel coordinates (u, v) in the camera image into the corresponding location on a bird's-eye-view heightmap defined in the robot's reference frame. Let $(u, v) \in \mathbb{Z}^2$ denote the pixel coordinates in the image and let $z = D(u, v)$ be the corresponding depth value (in meters), provided by a depth sensor with the same frame as the camera. The intrinsic matrix of the pinhole camera model is given by the simulation and is defined as:

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{3\times3} \tag{5.1}$$

The depth value provides the z-coordinate of the 3D point in the camera frame and the corresponding homogeneous image point is:

$$\mathbf{p}_{\text{pix}} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{5.2}$$

The 3D coordinates of the point in the camera frame are obtained by back-projecting the pixel using the inverse of the camera intrinsic matrix:

$$\mathbf{p}_{\text{cam}} = z \cdot K^{-1}\mathbf{p}_{\text{pix}} \in \mathbb{R}^3 \tag{5.3}$$

Next, the point is transformed into the world (robot) coordinate frame using the camera's given extrinsic matrix:

$$T_{\text{cam}}^{\text{world}} = \begin{bmatrix} R & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \in \mathbb{R}^{4\times4} \tag{5.4}$$

Here, $R \in \mathbb{R}^{3\times3}$ is the rotation matrix and $\mathbf{t} \in \mathbb{R}^3$ is the translation vector. The transformation is applied as:

$$\mathbf{p}_{\text{world}} = R \cdot \mathbf{p}_{\text{cam}} + \mathbf{t} \tag{5.5}$$

The resulting world coordinates (x, y, z) are then checked against the workspace limits: $[x_{\min}, x_{\max}]$, $[y_{\min}, y_{\max}]$, $[z_{\min}, z_{\max}]$.

If the point lies within this bounded workspace, it is projected to discrete coordinates in the top-down heightmap using a fixed resolution $\delta \in \mathbb{R}$, defined as meters per pixel. The pixel coordinates on the heightmap are computed as:

$$i = \left\lfloor \frac{x - x_{\min}}{\delta} \right\rfloor, \quad j = \left\lfloor \frac{y - y_{\min}}{\delta} \right\rfloor \tag{5.6}$$

These indices (i, j) correspond to the row and column in the 2D heightmap array where the projection of the 3D point is located.

Inverse Projection: From Heightmap to Camera View

We also implement the inverse process to map a spatial location defined in the top-down heightmap, constructed in the robot's reference frame, back into the 2D image plane of the camera. Given discrete indices $(i, j) \in \mathbb{Z}^2$ from the heightmap, the corresponding world coordinates (x, y) in meters are recovered using:

$$x = x_{\min} + i \cdot \delta, \quad y = y_{\min} + j \cdot \delta \tag{5.7}$$

where $\delta \in \mathbb{R}^+$ is the heightmap resolution (meters per pixel) and $[x_{\min}, x_{\max}], [y_{\min}, y_{\max}]$ are the workspace bounds along the respective axes. The z-coordinate is fixed as: $z = z_{\min}$ which corresponds to the surface level of the workspace.

Let

$$\mathbf{p}_{\text{world}} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \in \mathbb{R}^3 \tag{5.8}$$

denote the recovered 3D point in the robot/world coordinate frame. To map this to the camera coordinate system, the inverse of the camera pose is applied:

$$\mathbf{p}_{\text{cam}} = R^\top (\mathbf{p}_{\text{world}} - \mathbf{t}) \tag{5.9}$$

where $R \in \mathbb{R}^{3\times3}$ and $\mathbf{t} \in \mathbb{R}^3$ are again the rotation and translation components of the camera's extrinsic matrix $T_{\text{cam}}^{\text{world}}$. This effectively performs a rigid transformation from world to camera coordinates. Next, the camera projection model is applied to project this 3D point into the 2D image plane using the intrinsic matrix K:

$$\mathbf{p}_{\text{pix}} = K \cdot \mathbf{p}_{\text{cam}} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{\text{cam}} \\ y_{\text{cam}} \\ z_{\text{cam}} \end{bmatrix} \tag{5.10}$$

The resulting pixel coordinates (u, v) are obtained via homogeneous division:

$$v = \left\lfloor \frac{x_{\text{img}}}{z_{\text{cam}}} \right\rfloor, \quad u = \left\lfloor \frac{y_{\text{img}}}{z_{\text{cam}}} \right\rfloor \tag{5.11}$$

where u and v represent the row and column indices in the image.

Collecting trajectories heuristically: In the absence of learned models, simple yet effective heuristic methods can be employed to generate candidate robotic actions that cause movement to the objects in the environment based on geometric cues from the environment. Here, we iteratively rotate the depth heightmap across a fixed number of orientations. For each orientation, we identify potential pushing regions by computing where a forward shift of the rotated heightmap (in the intended push direction) reveals a significant drop in height. This depth discontinuity is interpreted as an opportunity to push an object, likely facilitating movement or separation of clustered items.

5.2 Training the Vision Model

The initial dataset contains 400 episodes, each with up to five randomly selected objects. The robot performs 10 pushes per episode and we record $r = 5$ frames per action. We train the vision model with the Adam optimizer [97] over 100 epochs using a multi-step learning rate schedule and $K = 10$ slots. As shown in Figure 5.4, the model effectively learns to bind distinct slots to different objects and reconstructs the input frames adequately well.

Different encoder architectures: We evaluate three different vision encoder architectures to study their impact on the performance of the Slot Attention module in our object-centric learning framework.

1. **Baseline Convolutional Encoder:** As a starting point, we adopt the convolutional encoder architecture proposed in the original Slot Attention paper [14]. This encoder consists of five convolutional layers with a relatively small receptive field. In our setup, however, we observe that the Slot Attention module struggles to learn meaningful object representations from the limited spatial context provided by this encoder, resulting in poor reconstruction and segmentation quality.
2. **Pretrained Vision Transformer (ViT):** To provide stronger object-centric features, we employ a Vision Transformer (ViT) pretrained with the DINO self-supervised learning method on ImageNet. The encoder consists of 12 transformer layers, each with 12 attention heads and a hidden dimensionality of 768. GELU [98] is used as the activation function. During training, the ViT weights are kept frozen and we append two trainable fully connected layers to adapt the extracted features before feeding them into the Slot Attention module. This configuration yields strong performance, allowing the model to detect objects robustly and reconstruct input images with high quality.
3. **Proposed ResNet-Based Encoder (Ours):** To eliminate reliance on external datasets and pretraining, we propose a ResNet-18 encoder with a large receptive field, trained entirely from scratch on our dataset. This encoder is trained jointly with the Slot Attention module using the reconstruction loss. Despite the lack of pretraining, we find that this architecture is capable of producing high-quality object representations and reconstructions when combined with Slot Attention.

In prior work on object-centric representation learning [15, 51, 90], the vision encoder is typically frozen and the object-awareness of DINO-pretrained models is leveraged to support slot-based decomposition. In contrast, our method successfully trains the vision encoder from scratch, enabling end-to-end learning without external visual priors. Nevertheless, we observe that training Slot Attention from scratch can be unstable and sensitive to initialization, consistent with findings in earlier work. We report a quantitative comparison of these encoder variants in Table 5.1.

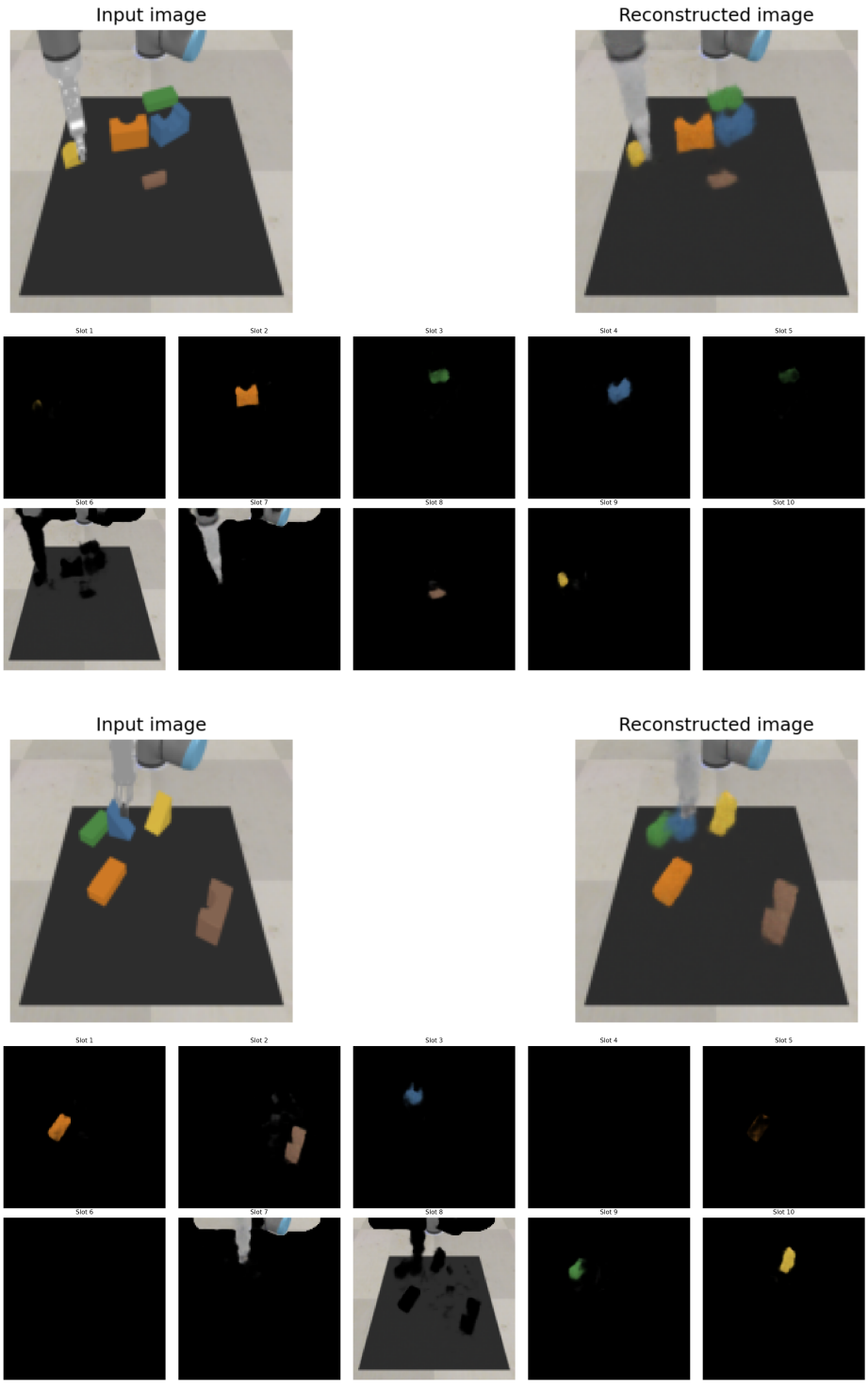


Figure 5.4: Training the vision model: Slot Attention learns to bind slots to individual objects and reconstructs relatively well the initial image. More than one slot might be bound to one object.

We observe that Slot Attention fails to converge to meaningful object representations when paired with a standard convolutional encoder. In this setup, the slots consistently collapse to modeling the background and no distinct object representations emerge. In contrast, both the pretrained Vision Transformer (ViT) and the ResNet-based encoder lead to significantly better performance.

The ViT encoder, pretrained with DINO, although inherently exhibits an object-centric bias, achieves slightly worse quantitative reconstruction performance than the ResNet encoder. We attribute this to the limited amount of training data available in our setting. Transformer-based architectures such as ViT are known to require large-scale datasets to fully realize their representational capacity. In contrast, the ResNet architecture, being convolutional and more inductive bias-driven, can perform more effectively under data constraints, making it better suited for low-data regimes like ours.

Encoder Architecture	**Test Data**
Simple CNN	0.4
Pretrained ViT	0.095
ResNet18	**0.073**

Table 5.1: **Evaluation of Different Vision Encoders** We report the reconstruction loss (MSE x10^{-2}) across three configurations: a CNN, a ViT and a ResNet18 based encoder.

Improving Masks with Entropy Regularization: At early stages of training, the segmentation masks produced by the Slot Attention module tend to be noisy, with multiple slots often attending to the same pixels. This overlap leads to ambiguous assignments and coarse object boundaries. To encourage sharper and more confident mask predictions, we introduce an entropy regularization term into the objective function, added alongside the standard reconstruction loss.

This regularization penalizes high entropy of the $K = 10$ attention masks, encouraging the model to assign each pixel more decisively to a single slot. Formally, we compute the pixel-wise entropy of the attention distributions across slots and minimize it, pushing the mask values closer to binary (i.e., 0 or 1). This constraint promotes finer segmentation and reduces overlap between slots. As illustrated in Figure 5.5, incorporating this loss significantly improves the quality of the produced masks by suppressing noise and yielding clearer, more object-aligned segmentations.

5.3 Training the World Model

For the world model, we adopt a training procedure similar to that of [20], keeping the vision components (encoder, Slot Attention and decoder) frozen. The world model takes as input the

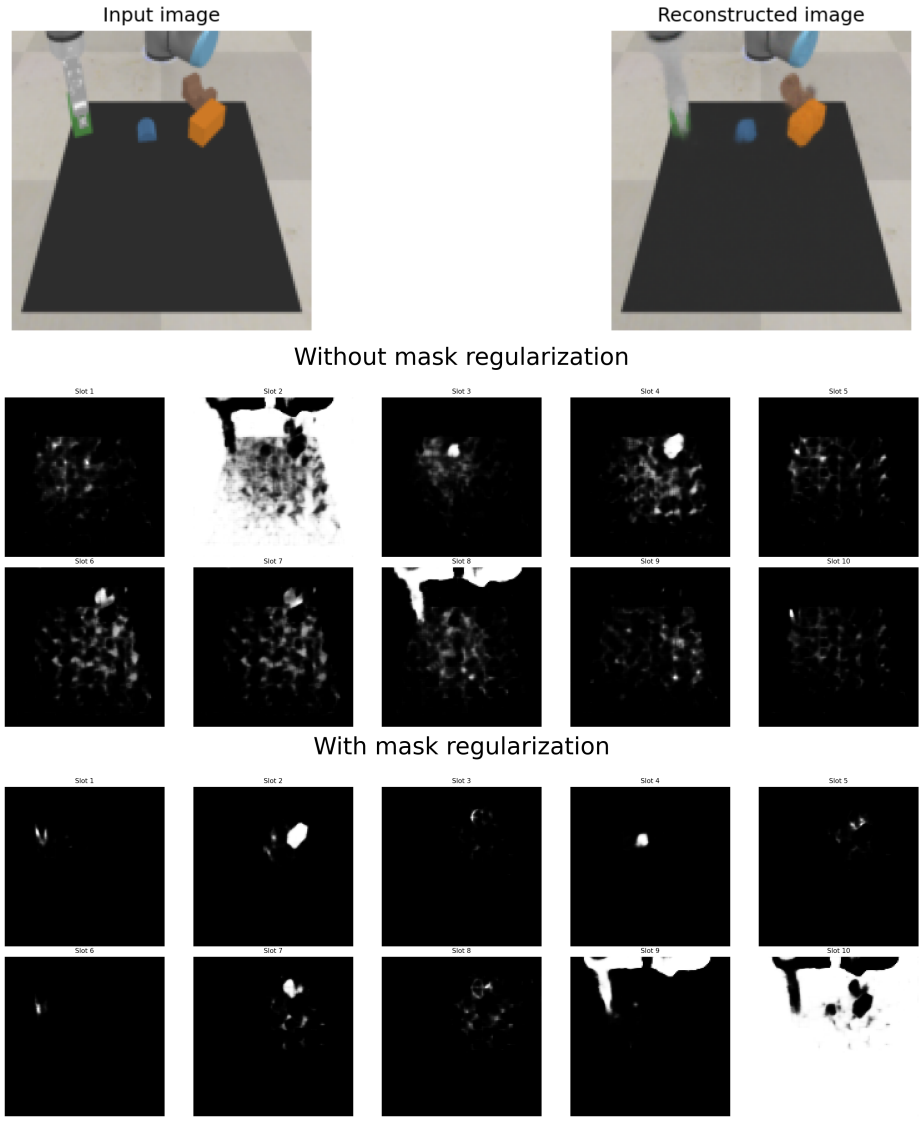


Figure 5.5: Effect of entropy-based mask regularization. When the entropy regularization term is added to the loss function, the resulting segmentation masks become sharper and more aligned with object boundaries. This reduces noise caused by overlapping slot attention across pixels, resulting in clearer and more consistent segmentation.

$K = 10$ slots of dimension $D = 128$ produced by the frozen encoder and Slot Attention module. The GNN takes these vectors along with the 3-dimensional action vector and predicts the latent object representations at the next time step. The action vector is first processed by a two-layer MLP, yielding an action embedding of size 16 that is broadcast and concatenated to each object's latent feature.

The GNN comprises two main components: an edge MLP and a node MLP. The edge model processes every pair of object embeddings, producing 128-dimensional interaction messages using a 3-layer MLP. These edge features are aggregated for each node. The aggregated message, the original node embedding and the processed action are then concatenated and passed through the node MLP, also a 3-layer network. The output is a set of updated latent vectors of the same dimension (128). Training is done with the Adam optimizer on the objective described in Section 4.3

We test the proposed method by training the world model on trajectories collected under heuristic pushing actions that cause movement. As shown in Figure 5.6 the world model manages to predict the orientation of the pushed object's movement.

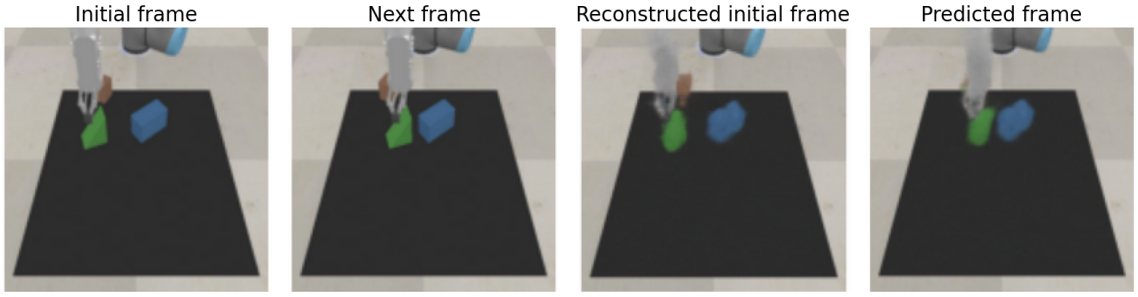


Figure 5.6: The world model is able to predict the movement of the green object, when trained on the dataset comprised of heuristic actions.

Now we train the world model using the trajectories collected under randomly chosen pushing actions. However, due to the limited diversity of the dataset, the world model struggles to develop a meaningful understanding of object dynamics. In most episodes, the only moving element is the gripper, which biases the model toward learning only *ego-motion*. As a result, it fails to capture or predict the movement of objects when interacted with, as illustrated in Figure 5.7.

5.4 Designing an Intrinsic Reward Function

With a functional understanding of the environment in place, we proceed to actively collect informative data. At this stage, the vision model is capable of reasonably localizing objects and the

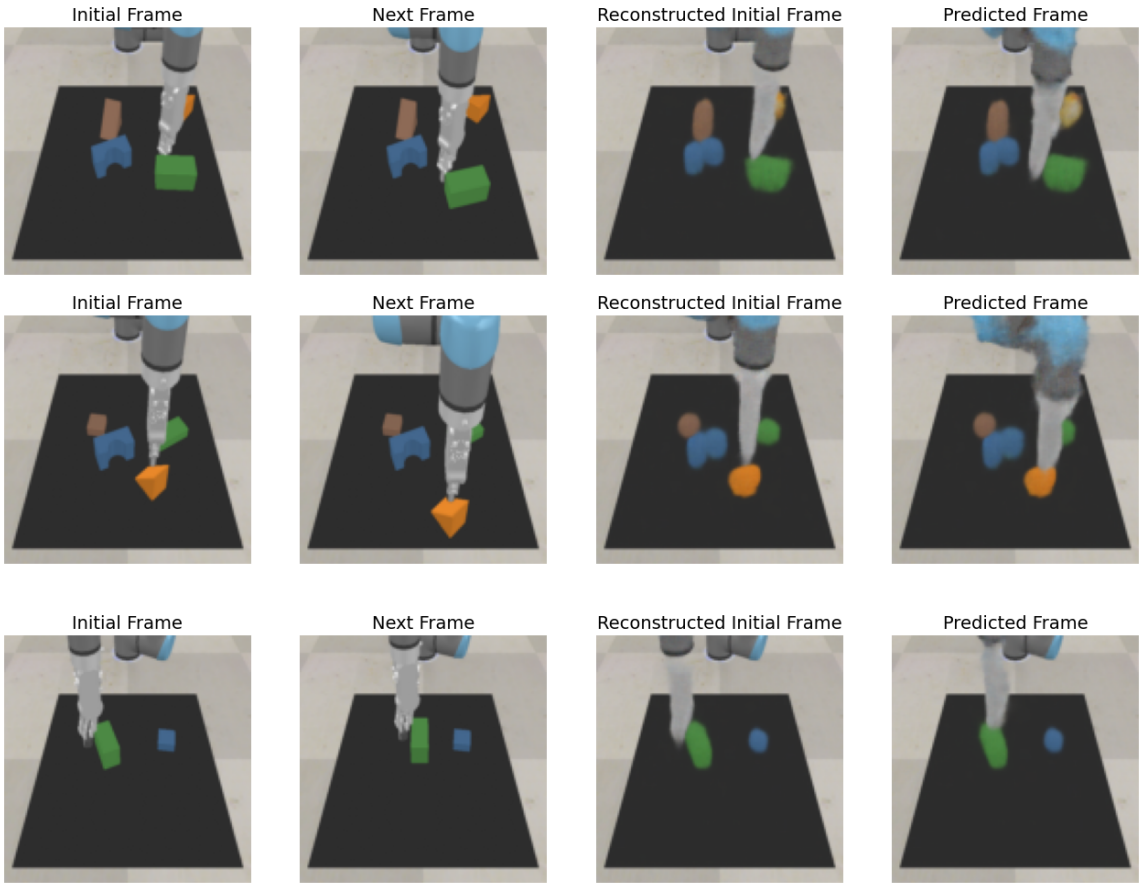


Figure 5.7: When trained on the random dataset the world model can only predict ego-motion

world model can predict the gripper's motion with fair accuracy. These capabilities are sufficient to drive a policy that seeks novel and informative trajectories.

We begin by performing a statistical analysis of the different intrinsic reward signals introduced in Section 4.4: (i) the reconstruction loss of the vision model, (ii) the prediction error of the world model and (iii) the variance among predictions from an ensemble of world models.

To evaluate these signals, we compare their values under the two types of actions: (1) heuristic actions designed to cause object movement and (2) random actions that may or may not cause such movement. Our goal is to test the hypothesis that moving actions result in significantly different reward signals compared to non-moving actions, which would indicate that a given reward function can effectively drive exploration.

Assuming that the reward signals are normally distributed with equal variance, we apply an independent t-test to evaluate statistical significance. The results show that both the reconstruction loss and the world model prediction error exhibit significantly higher values under movement-inducing actions. This indicates their potential as intrinsic rewards that encourage interaction with objects. In contrast, the variance among ensemble predictions does not show a significant difference, suggesting it may be less reliable for this purpose. The distributions of these signals are visualized in Figure 5.8.

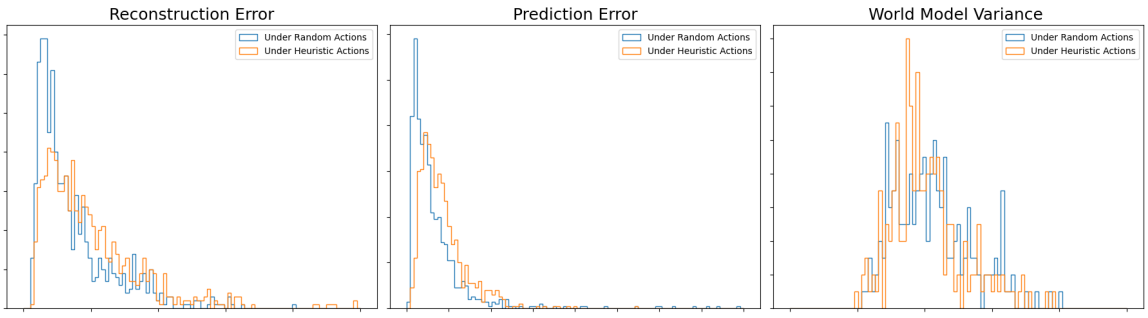


Figure 5.8: The distribution of the different reward function under random and under heuristic actions. Both reconstruction error and world model prediction error seem to be able to serve as a reward.

Despite these findings, we observe empirically that using reconstruction loss or prediction error alone results in noisy reward signals that are insufficient to robustly guide policy learning. Consequently, we focus on the prediction error-based reward and enhance it by incorporating two key modifications: (i) a reference error baseline and (ii) a spatial mask that localizes the reward around the action point, as described in Section 4.4.

The intrinsic reward was computed online during training using the prediction error from the world model. At each time step, the model decoded both the predicted and current latent states to

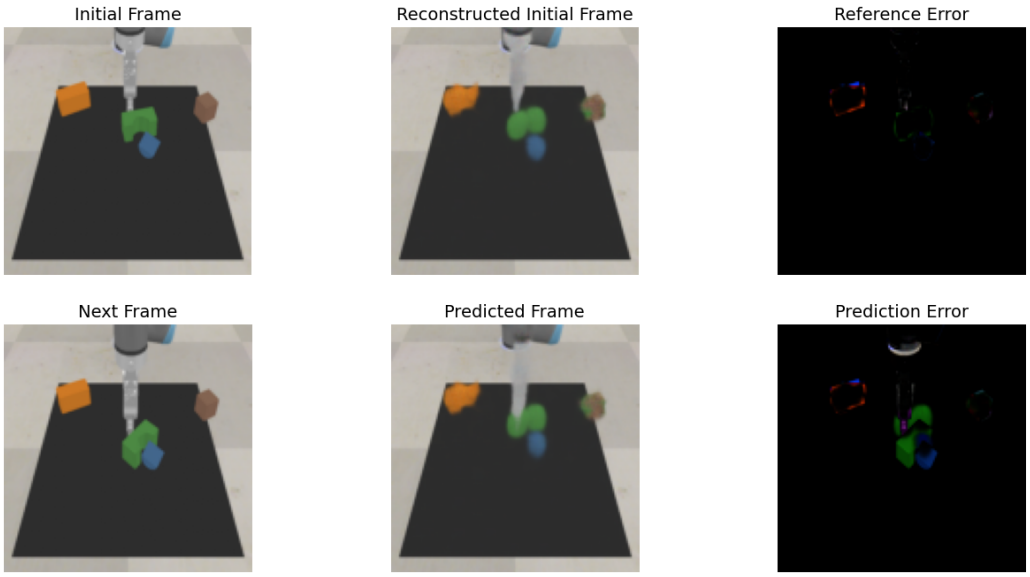


Figure 5.9: Designing the reward function. The world model exhibits high prediction error where object motion occurs, particularly for the green object affected by the robotic arm. By subtracting the reference reconstruction error, we isolate the action-dependent prediction error and suppress irrelevant errors such as those from the static orange object, which is not influenced by the action.

produce future and current frame reconstructions, respectively. The reward was evaluated over a spatial mask of radius $d = 15$ pixels, centered at the action point in image space. In Figures 5.9, 5.10 we visualize the intuition behind the designing of the reward function.

5.5 Collecting Informative Trajectories

Our intrinsic reward signal, designed to filter out noise from the partially trained vision and world models, successfully reflects the world model's predictive uncertainty. The policy is trained using Double Q-learning [36], for 1000 steps, with a classic decaying epsilon-greedy exploration strategy. The Q-network is a ResNet that takes as input the 10 segmentation masks $\{\mathbf{\Pi}_k\}$ concatenated in the channel dimension. To obtain the proposed action we randomly choose from $N = 100$ pixels with the highest value in the map that the Q Network outputs and then transform the pixel to an exact location in the robot's reference frame as explained in previous sections.

Interestingly, the learned policy predominantly suggests pushing actions that cause *object motion*. We quantify this observation by measuring that the average displacement of object centers

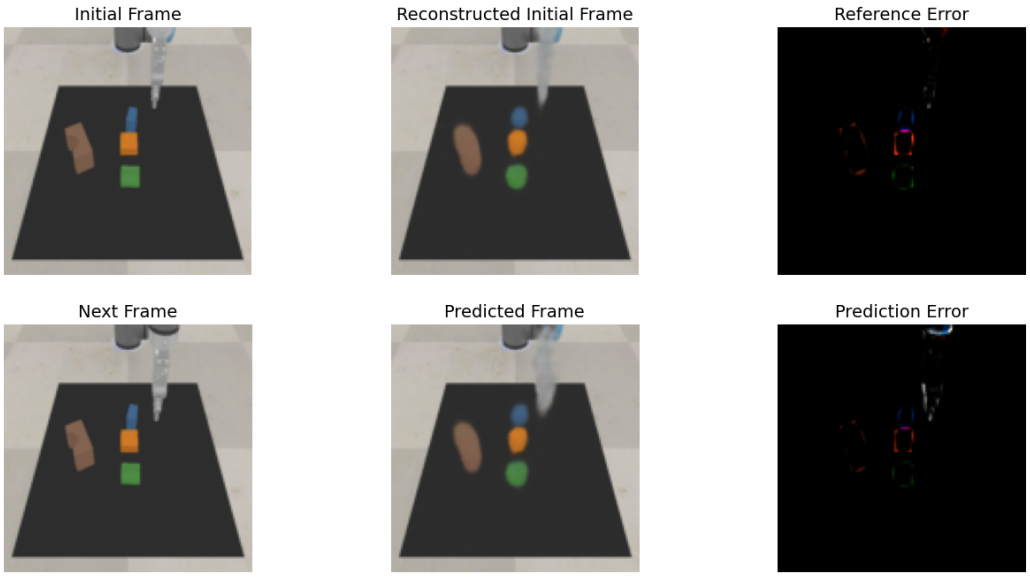


Figure 5.10: Handling static scenes. When the action does not induce object motion, the overall prediction error is dominated by reconstruction noise from the vision model. Subtracting the reference error ensures that such noise does not contribute to the reward, resulting in minimal intrinsic reward in static scenes.

of mass is **three times greater** than under random actions. More specifically, the simulation environment provides access to the 3D positions of the centers of mass for all objects in the scene. We conduct 100 episodes under a random action policy and 100 episodes under our learned policy, recording the average displacement of objects per episode. This analysis reveals that the learned policy more frequently selects actions that result in meaningful object interaction and motion. We also demonstrate this qualitatively in Figure 5.11 by visualizing the positions of the best actions that the learned policy suggests. These visualizations show a clear focus on areas near or in contact with objects, in contrast to the uniform and object-agnostic distribution of random actions.

5.6 Enhancing World Perception and Modeling

We leverage the learned, intrinsically motivated policy to collect a new dataset composed of more diverse and dynamic interactions, particularly involving object motion. As before, the new dataset contains 400 episodes of 10 actions each. We initialize the vision and world models with parameters from the initial training and fine-tune them for 100 epochs using a reduced learning rate and the same training setup.

As shown in Tables 5.2, 5.3, fine-tuning on this enriched dataset leads to substantial improvements in both reconstruction and predictive accuracy, compared to models trained exclusively

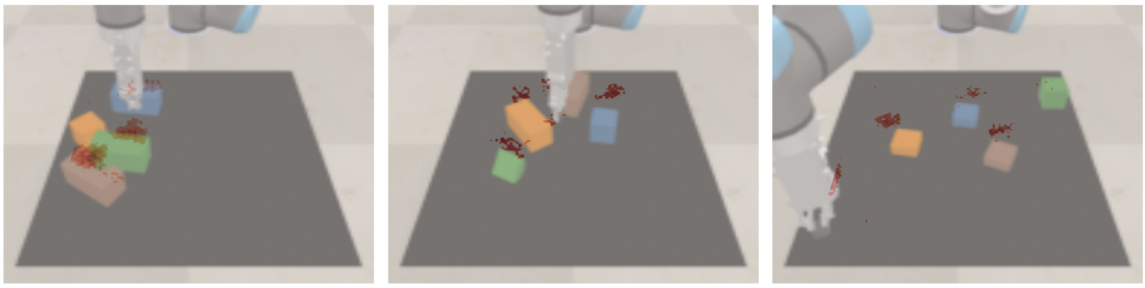


Figure 5.11: **Policy visualization:** We visualize the output of the Q-network by marking as red the pixels that our policy suggests. The red pixels clearly illustrate that the agent has learned to prefer actions that push objects.

on data from random actions. To showcase this, we evaluate the models on the control dataset; namely on newly, independent, collected test trajectories generated using either random or heuristic action policies. In more detail, Table 5.2 shows that the re-trained vision encoder provides more balanced results across the test sets, compared to the action-biased alternatives. The vision model can now reconstruct better the instances corresponding to heuristic actions, while maintaining its performance on instances of random actions. In practice, we want good reconstruction to both random and heuristic data to reflect our ability to "perceive" our world.

Moreover, in the fourth column of Figure 5.12 we demonstrate that the world model is now able to predict the movement of the object when pushed by the gripper, which is quantitatively shown in Table 5.3. Here, we consider two enhancement pipelines: 1) keep the initial vision encoder frozen and fine-tune the world model and 2) use the previously enhanced vision (EV) encoder frozen and fine-tune the world model. As expected, the enhanced vision encoder further improves the reconstruction metric. Note that both enhancement versions, provide better results compared to the considered biased alternatives, even with respect to the system trained on heuristic actions.

Trained On	**Random Test Data**	**Heuristic Test Data**
Random Actions	**0.062**	0.084
Heuristic Actions	0.070	**0.053**
Intrinsic Reward Policy	0.063	0.069

Table 5.2: **Evaluation of Vision Models on Different Test Datasets** We report the reconstruction loss (MSE x10^{-2}) across three configurations. A vision model trained on random actions, one trained on heuristic actions and one trained on the dataset collected using the intrinsically motivated policy.

Trained On	Random Test Data	Heuristic Test Data
Random Actions	0.209	0.235
Heuristic Actions	0.247	0.164
Intrinsic Reward Policy	**0.131**	0.143
Intrinsic Reward Policy (EV)	**0.131**	**0.135**

Table 5.3: **Evaluation of World Models on Different Test Datasets** We report the reconstruction loss (MSE x10^{-2}) for the predicted next frame across four configurations: a world model trained on random actions, one trained on heuristic actions and two trained on the dataset collected using the intrinsically motivated policy, either with the initial vision model or the enhanced vision model (EV).

5.7 Discussion

Our experiments demonstrate that world perception and understanding can emerge when an agent, equipped with a self-supervised object-centric framework and driven by intrinsic motivation, leverages its ability to interact with its environment. A key finding is that the prediction error of the world model induces meaningful object motion. This behavior arises without any external supervision, supporting the hypothesis that model uncertainty can effectively guide exploration.

We also observe that training a vision model in a fully self-supervised manner with limited data remains a significant challenge. The Slot Attention module fails to produce meaningful segmentations when paired with a shallow CNN encoder and successful convergence requires careful hyperparameter tuning. Interestingly, although the DINO-pretrained Vision Transformer exhibits strong object-centric priors, its performance is slightly worse than that of the ResNet-based encoder trained from scratch, likely due to the limited size of our dataset and the superior data efficiency of convolutional architectures.

We demonstrate that the quality and variety of collected data are highly correlated with the world model's ability to accurately capture object dynamics. When trained solely on randomly collected episodes, the model learns little beyond gripper motion. In contrast, trajectories gathered via heuristic or learned policies enable the model to capture and predict object interactions.

Crucially, we find that fine-tuning both the vision and world models on data collected through intrinsically motivated exploration leads to measurable improvements in both reconstruction and prediction. The vision model trained on this dataset achieves a better overall reconstruction error, while the world model performs best when both it and the vision encoder are fine-tuned on the more informative data.

Our work operates in a setting without external supervision, there are no object labels, bound-

ing boxes, or class annotations, making traditional supervised methods for instance segmentation like Mask R-CNN [99] or YOLO [100] unsuitable. One alternative could be leveraging self-supervised methods such as Masked Autoencoders (MAE) [**?**]. In principle, we could pretrain a MAE-based Vision Transformer on our dataset and use its features both for world modeling and as a state representation for the reinforcement learning pipeline. However, in practice, this approach proved unstable. Due to the limited visual diversity in our dataset, MAE training failed to converge to meaningful representations. Moreover, MAE models are typically pretrained on large-scale, object-rich datasets like ImageNet, which our environment does not resemble. In contrast, we chose to use Slot Attention, which introduces a strong object-centric inductive bias and allows for self-supervised object segmentation. This design enables us to train the world model directly on structured object representations and to use the corresponding segmentation masks as interpretable input to the policy network.

This approach offers several practical advantages. First, it brings interpretability: each slot corresponds to a specific object, and when an action affects an object, the world model learns to update the corresponding slot embedding. Second, world model training becomes simpler, as the model only needs to predict object-level embeddings, rather than high-dimensional, entangled features. Third, the segmentation masks output by Slot Attention provide a spatial structure that makes it easier for the policy network to learn where to act. Compared to learning policies over dense MAE feature maps, using discrete object masks helps focus exploration and simplifies the learning problem. That said, Slot Attention comes with its own drawbacks. Training it from scratch is notoriously fragile and requires careful hyperparameter tuning and architectural choices. Despite its popularity in the vision community, it is not yet a plug-and-play solution. Still, our results suggest that object-centric representation learning, especially in low-data regimes, is a promising research direction with open challenges and strong potential for impact in real-world robotic settings.

Overall, our experiments validate the central idea that a self-supervised agent, guided by intrinsic incentives, can incrementally construct more effective models of its environment. While our approach has limitations, such as training instability and the relative simplicity of the simulated environment, it still provides strong evidence that cognition-inspired principles like curiosity, object-centric perception, and predictive world modeling can significantly enhance artificial systems.

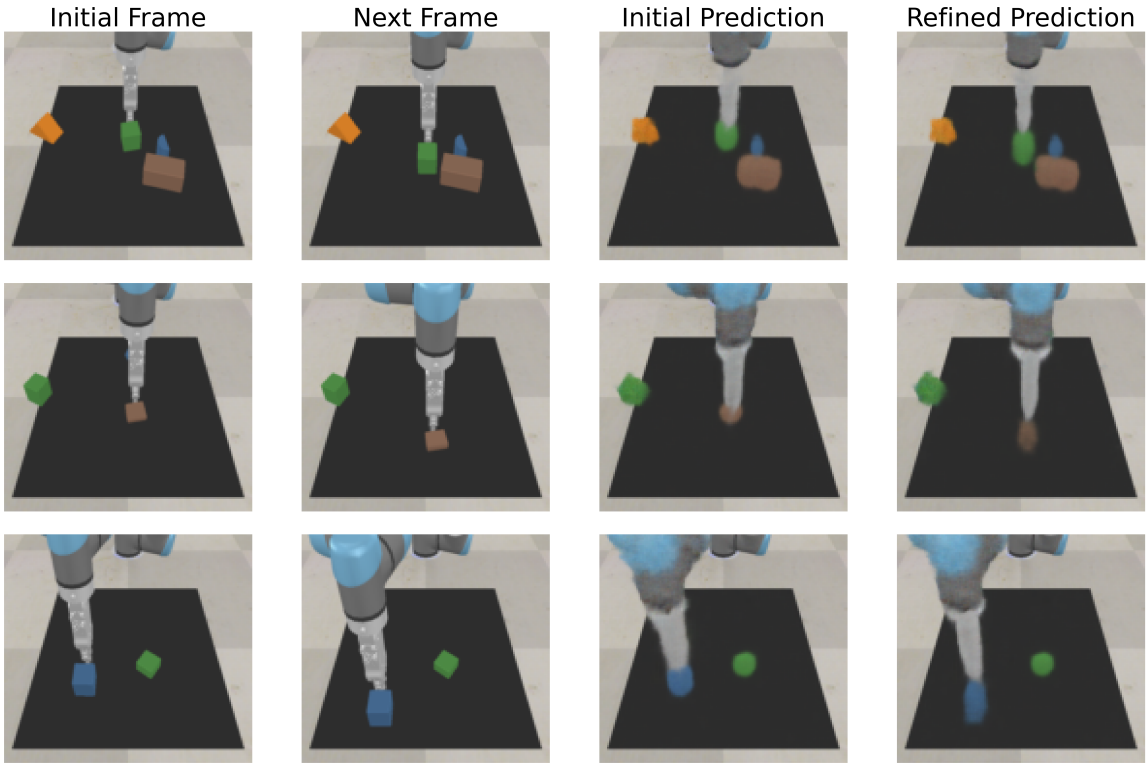


Figure 5.12: **Qualitative Evaluation:** The first and second columns show image frames before and after an action. The third and fourth columns display the predicted motion from the world models trained on the initial dataset and on the new, informative trajectories, respectively. Note the improved accuracy of the refined model in capturing object movement.

Chapter 6

Conclusion

Contents

6.1 Summary and Further Discussion

This work explores whether a robot can adequately perceive its surrounding and their dynamics purely through interaction, without relying on prior knowledge or external supervision. Drawing inspiration from cognitive science, which suggests that perception is shaped by internal predictive models and that intelligence emerges from continuous sensorimotor experience [69, 24], we looked to emulate similar mechanisms in artificial agents. Human infants, for example, learn incrementally through physical interaction with their environment, with objects playing a fundamental role in how they structure and interpret visual input [72].

Building on these principles, we proposed a self-supervised, object-centric learning framework that enables a robot to incrementally improve its perceptual and predictive capabilities through intrinsically motivated exploration. By fine-tuning both the vision and world models on data collected through a learned policy, our system significantly improved in terms of both reconstruction quality and dynamics prediction accuracy.

One of the primary challenges we faced was the absence of external supervision. Modern deep learning models typically excel when trained on large, curated datasets. In contrast, training a vision model to extract useful object-centric representations from limited, noisy and unstructured data proved difficult. In particular, ensuring the convergence of Slot Attention was non-trivial and required careful tuning of hyperparameters, weight initialization schemes and the use of auxiliary loss functions, as discussed in previous sections. These observations reinforce the broader understanding that deep learning models still face substantial limitations in low-data, self-supervised settings. Consequently, contributions such as ours, which propose ways to incrementally improve model performance in the absence of supervision, are essential steps forward.

Another significant challenge was the design of a reward function capable of driving meaningful and coherent policies. While using model prediction errors as a source of intrinsic motivation is conceptually straightforward, it is practically difficult. When models are trained on limited or uninformative data, their representations are often noisy and unreliable. To address this, we designed a reward function that filters out vision model noise and focuses primarily on errors in predicted dynamics. Nevertheless, in more realistic or complex environments, stronger vision models with better generalization capabilities may be required to support reliable reward signals and robust policy learning.

In conclusion, our work demonstrates two important insights. First, principles from cognitive science and developmental psychology can inspire effective learning frameworks for autonomous agents. Second, it underscores the current limitations of deep learning in enabling fully autonomous, self-supervised learning in open-ended, real-world environments.

6.2 Future Work

A natural extension of this thesis is to explore continual learning [70]. In continual learning, models are updated incrementally as new data becomes available, allowing them to adapt over time without retraining from scratch. In our context, this could be implemented by alternating between two stages: (1) a data collection stage, where the agent interacts with the environment using a policy guided by a reward function shaped by the vision and world models and (2) a learning stage, where the vision and world models are updated based on the newly collected data. This framework would raise questions about how to mitigate catastrophic forgetting and ensure long-term knowledge retention, two core challenges in continual learning research.

Another promising direction involves a deeper exploration of the quality and utility of the learned object-centric representations. In this thesis, we primarily use these representations to improve the formulation of the world model. However, they may also encode rich information about object attributes, such as shape, position, or affordances, that could be leveraged in downstream control tasks. An immediate application would be to train a policy directly on the object-centric representations and evaluate whether this abstraction facilitates faster or more robust skill acquisition compared to raw pixel-based inputs.

Our framework also introduces an implicit mechanism for evaluating world models qualitatively. By observing the behavior of the trained policy, particularly how the robot interacts with objects of varying shapes and configurations, we can infer properties of the underlying world model. For example, a policy that adapts its behavior to different object geometries may reflect a world model that has successfully captured key physical dynamics.

Finally, an ambitious direction for future work is to implement our method in a real-world robotic setting. In a basic version, a robotic arm could interact with objects while a human periodically rearranges the scene to introduce novelty. However, this could evolve into a fully autonomous scenario involving navigation and interaction. In this setting the robot would have an extended action space that includes locomotion, facilitating the ability to choose whether or not to engage with objects at specific locations. Initially, the robot would collect data through random interactions. It would then use this data to train its vision and world models. These models, in turn, would be used to shape an intrinsic reward signal, driving further exploration in ways that yield improved perception and understanding of both the visual scene and the underlying dynamics. With each iteration, the robot would refine its internal models using the richer data generated under the updated policy.

6.3 Towards Cognitively Inspired Machine Learning

Artificial intelligence has long been influenced by insights from natural cognition. For example, the nodes in artificial neural networks are loosely inspired by the behavior of biological neurons, convolutional operations in CNNs were motivated by early studies of the visual cortex in cats and Transformer architectures are built around the concept of attention, a fundamental mechanism in human cognition.

This ongoing dialogue between neuroscience, cognitive science, developmental psychology and machine learning has proven mutually beneficial. Machine learning continues to draw inspiration from these fields to design more robust and intelligent systems, while cognitive and brain sciences increasingly leverage machine learning techniques to model and understand the complexities of natural intelligence.

We believe that this thesis represents another contribution to this interdisciplinary exchange. By drawing on cognitive principles, such as intrinsic motivation, object-centric perception and self-supervised learning through interaction, we offer not only a pathway to more reliable, autonomous and adaptive artificial systems, but also a framework that could inform and support theories of how learning unfolds in biological agents.

Bibliography

[1] O. Konstantaropoulos, M. Khamassi, P. Maragos, and G. Retsinas, "Push, see, predict: Emergent perception through intrinsically motivated play," in *Proceedings of the IEEE International Conference on Development and Learning (ICDL)*, 2025.

[2] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15979–15988, 2022.

[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–44, 05 2015.

[4] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[5] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *ArXiv*, vol. abs/2010.11929, 2020.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), vol. 30, Curran Associates, Inc., 2017.

[7] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, (Red Hook, NY, USA), p. 1025–1035, Curran Associates Inc., 2017.

[8] M. Caron, H. Touvron, I. Misra, H. Jegou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9630–9640, 2021.

[9] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction.* Cambridge, MA, USA: A Bradford Book, 2018.

[10] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2017.

[11] S. Caelles, J. Pont-Tuset, F. Perazzi, A. Montes, K.-K. Maninis, and L. V. Gool, "The 2019 davis challenge on vos: Unsupervised multi-object segmentation," *ArXiv*, vol. abs/1905.00737, 2019.

[12] C. Yang, H. Lamdouar, E. Lu, A. Zisserman, and W. Xie, "Self-supervised video object segmentation by motion grouping," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7157–7168, 2021.

[13] M. Engelcke, A. R. Kosiorek, O. P. Jones, and I. Posner, "GENESIS: generative scene inference and sampling with object-centric latent representations," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, OpenReview.net, 2020.

[14] F. Locatello, D. Weissenborn, T. Unterthiner, A. Mahendran, G. Heigold, J. Uszkoreit, A. Dosovitskiy, and T. Kipf, "Object-centric learning with slot attention," in *Advances in Neural Information Processing Systems* (H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, eds.), vol. 33, pp. 11525–11538, Curran Associates, Inc., 2020.

[15] M. Seitzer, M. Horn, A. Zadaianchuk, D. Zietlow, T. Xiao, C.-J. Simon-Gabriel, T. He, Z. Zhang, B. Schölkopf, T. Brox, and F. Locatello, "Bridging the gap to real-world object-centric learning," in *The Eleventh International Conference on Learning Representations*, 2023.

[16] A. Zadaianchuk, M. Seitzer, and G. Martius, "Object-centric learning for real-world videos by predicting temporal feature similarities," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[17] K. Fan, Z. Bai, T. Xiao, T. He, M. Horn, Y. Fu, F. Locatello, and Z. Zhang, "Adaptive slot attention: Object discovery with dynamic slot number," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23062–23071, 2024.

[18] S. Levine, P. Pastor, A. Krizhevsky, and D. Quillen, "Learning hand-eye coordination for robotic grasping with large-scale data collection," in *International Symposium on Experimental Robotics*, 2016.

[19] A. Zadaianchuk, M. Seitzer, and G. Martius, "Self-supervised visual reinforcement learning with object-centric representations," in *International Conference on Learning Representations*, 2021.

[20] T. Kipf, E. van der Pol, and M. Welling, "Contrastive learning of structured world models," in *International Conference on Learning Representations*, 2020.

[21] Y. Seo, D. Hafner, H. Liu, F. Liu, S. James, K. Lee, and P. Abbeel, "Masked world models for visual control," in *6th Annual Conference on Robot Learning*, 2022.

[22] A. Gopnik, "Scientific thinking in young children: Theoretical advances, empirical research, and policy implications," *Science*, vol. 337, no. 6102, pp. 1623–1627, 2012.

[23] C. Kidd and B. Y. Hayden, "The psychology and neuroscience of curiosity," *Neuron*, vol. 88, no. 3, pp. 449–460, 2015.

[24] L. B. Smith and M. Gasser, "The development of embodied cognition: Six lessons from babies," *Artificial Life*, vol. 11, pp. 13–29, 2005.

[25] D. Ha and J. Schmidhuber, "World models.," *CoRR*, vol. abs/1803.10122, 2018.

[26] Y. Matsuo, Y. LeCun, M. Sahani, D. Precup, D. Silver, M. Sugiyama, E. Uchibe, and J. Morimoto, "Deep learning, reinforcement learning, and world models," *Neural Networks*, vol. 152, pp. 267–275, 2022.

[27] Y. LeCun, "A path towards autonomous machine intelligence," 2022.

[28] A. Baranes and P.-Y. Oudeyer, "Active learning of inverse models with intrinsically motivated goal exploration in robots," *Robotics and Autonomous Systems*, vol. 61, no. 1, pp. 49–73, 2013.

[29] V. G. Santucci, G. Baldassarre, and M. Mirolli, "Grail: a goal-discovering robotic architecture for intrinsically-motivated learning," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 8, no. 3, pp. 214–231, 2016.

[30] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. USA: Prentice Hall Press, 3rd ed., 2009.

[31] I. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, M. Lucic, and A. Dosovitskiy, "Mlp-mixer: an all-mlp architecture for vision," in *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, (Red Hook, NY, USA), Curran Associates Inc., 2021.

[32] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-dujaili, Y. Duan, O. Al-Shamma, J. I. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: concepts, cnn architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, 2021.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[34] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

[35] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller, "Playing atari with deep reinforcement learning," *CoRR*, vol. abs/1312.5602, 2013.

[36] H. v. Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, p. 2094–2100, AAAI Press, 2016.

[37] Y. Wang, U. Ahsan, H. Li, and M. Hagen, "A comprehensive review of modern object segmentation approaches," *Found. Trends. Comput. Graph. Vis.*, vol. 13, p. 111–283, Oct. 2022.

[38] I. Biederman, "Human image understanding: Recent research and a theory," *Computer Vision, Graphics, and Image Processing*, vol. 32, no. 1, pp. 29–73, 1985.

[39] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 724–732, 2016.

[40] P. Ochs, J. Malik, and T. Brox, "Segmentation of moving objects by long term video analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 6, pp. 1187–1200, 2014.

[41] K. Fragkiadaki, P. Arbelaez, P. Felsen, and J. Malik, " Learning to segment moving objects in videos ," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Los Alamitos, CA, USA), pp. 4083–4090, IEEE Computer Society, June 2015.

[42] H. Song, W. Wang, S. Zhao, J. Shen, and K.-M. Lam, "Pyramid dilated deeper convlstm for video salient object detection," in *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XI*, (Berlin, Heidelberg), p. 744–760, Springer-Verlag, 2018.

[43] X. Lu, W. Wang, C. Ma, J. Shen, L. Shao, and F. Porikli, "See more, know more: Unsupervised video object segmentation with co-attention siamese networks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3618–3627, 2019.

[44] M. Lee, S. Cho, S. Lee, C. Park, and S. Lee, "Unsupervised video object segmentation via prototype memory network," in *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 5913–5923, 2023.

[45] C. P. Burgess, L. Matthey, N. Watters, R. Kabra, I. Higgins, M. M. Botvinick, and A. Lerchner, "Monet: Unsupervised scene decomposition and representation," *CoRR*, vol. abs/1901.11390, 2019.

[46] K. Greff, R. L. Kaufman, R. Kabra, N. Watters, C. Burgess, D. Zoran, L. Matthey, M. M. Botvinick, and A. Lerchner, "Multi-object representation learning with iterative variational inference," in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA* (K. Chaudhuri and R. Salakhutdinov, eds.), vol. 97 of *Proceedings of Machine Learning Research*, pp. 2424–2433, PMLR, 2019.

[47] A. R. Kosiorek, H. Kim, Y. W. Teh, and I. Posner, "Sequential attend, infer, repeat: Generative modelling of moving objects," in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada* (S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds.), pp. 8615–8625, 2018.

[48] J. Jiang, S. Janghorbani, G. de Melo, and S. Ahn, "SCALOR: generative world models with scalable object representations," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, OpenReview.net, 2020.

[49] Z. Lin, Y. Wu, S. V. Peri, W. Sun, G. Singh, F. Deng, J. Jiang, and S. Ahn, "SPACE: unsupervised object-oriented scene representation via spatial attention and decomposition," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, OpenReview.net, 2020.

[50] Y. Wu, O. Parker Jones, M. Engelcke, and I. Posner, "Apex: Unsupervised, object-centric scene segmentation and tracking for robot manipulation," pp. 3375–3382, 09 2021.

[51] A. Zadaianchuk, M. Kleindessner, Y. Zhu, F. Locatello, and T. Brox, "Unsupervised semantic segmentation with self-supervised object-centric representations," in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*, OpenReview.net, 2023.

[52] P. Agrawal, A. Nair, P. Abbeel, J. Malik, and S. Levine, "Learning to poke by poking: experiential learning of intuitive physics," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, (Red Hook, NY, USA), p. 5092–5100, Curran Associates Inc., 2016.

[53] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, and S. Levine, "Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation," *CoRR*, vol. abs/1806.10293, 2018.

[54] E. Jang, C. Devin, V. Vanhoucke, and S. Levine, "Grasp2vec: Learning object representations from self-supervised grasping," in *2nd Annual Conference on Robot Learning, CoRL 2018, Zürich, Switzerland, 29-31 October 2018, Proceedings*, vol. 87 of *Proceedings of Machine Learning Research*, pp. 99–112, PMLR, 2018.

[55] N. Heravi, A. Wahid, C. Lynch, P. Florence, T. Armstrong, J. Tompson, P. Sermanet, J. Bohg, and D. Dwibedi, "Visuomotor control in multi-object scenes using object-aware representations," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9515–9522, 2023.

[56] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, "Curiosity-driven exploration by self-supervised prediction," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, p. 2778–2787, JMLR.org, 2017.

[57] Y. Burda, H. Edwards, D. Pathak, A. Storkey, T. Darrell, and A. A. Efros, "Large-scale study of curiosity-driven learning," in *International Conference on Learning Representations*, 2019.

[58] D. Pathak, D. Gandhi, and A. Gupta, "Self-supervised exploration via disagreement," in *Proceedings of the 36th International Conference on Machine Learning* (K. Chaudhuri and R. Salakhutdinov, eds.), vol. 97 of *Proceedings of Machine Learning Research*, pp. 5062–5071, PMLR, 09–15 Jun 2019.

[59] H. Liu and P. Abbeel, "Behavior from the void: Unsupervised active pre-training," in *Advances in Neural Information Processing Systems* (A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, eds.), 2021.

[60] L. Pinto, D. Gandhi, Y. Han, Y.-L. Park, and A. Gupta, "The curious robot: Learning visual representations via physical interactions," in *Computer Vision – ECCV 2016* (B. Leibe, J. Matas, N. Sebe, and M. Welling, eds.), (Cham), pp. 3–18, Springer International Publishing, 2016.

[61] D. Pathak, Y. Shentu, D. Chen, P. Agrawal, T. Darrell, S. Levine, and J. Malik, "Learning instance segmentation by interaction," in *2018 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 2042–2045, Computer Vision Foundation / IEEE Computer Society, 2018.

[62] C. Sancaktar, S. Blaes, and G. Martius, "Curious exploration via structured world models yields zero-shot object manipulation," in *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022* (S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, eds.), 2022.

[63] N. Watters, L. Matthey, M. Bosnjak, C. P. Burgess, and A. Lerchner, "COBRA: data-efficient model-based RL through unsupervised object discovery and curiosity-driven exploration," *CoRR*, vol. abs/1905.09275, 2019.

[64] K. Cho, B. van Merrienboer, Çaglar Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," in *Conference on Empirical Methods in Natural Language Processing*, 2014.

[65] N. Watters, L. Matthey, C. P. Burgess, and A. Lerchner, "Spatial broadcast decoder: A simple architecture for learning disentangled representations in vaes," *CoRR*, vol. abs/1901.07017, 2019.

[66] M. Caron, H. Touvron, I. Misra, H. J'egou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9630–9640, 2021.

[67] T. Kipf, G. F. Elsayed, A. Mahendran, A. Stone, S. Sabour, G. Heigold, R. Jonschkowski, A. Dosovitskiy, and K. Greff, "Conditional object-centric learning from video," in *The Tenth*

International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022, OpenReview.net, 2022.

[68] E. Rohmer, S. P. N. Singh, and M. Freese, "Coppeliasim (formerly v-rep): a versatile and scalable robot simulation framework," in *Proc. of The International Conference on Intelligent Robots and Systems (IROS)*, 2013. www.coppeliarobotics.com.

[69] N. Nortmann, S. Rekauzke, S. Onat, P. König, and D. Jancke, "Primary visual cortex represents the difference between past and present," *Cerebral Cortex (New York, NY)*, vol. 25, pp. 1427 – 1440, 2013.

[70] L. Wang, X. Zhang, H. Su, and J. Zhu, "A comprehensive survey of continual learning: Theory, method and application," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 8, pp. 5362–5383, 2024.

[71] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012.

[72] E. S. Spelke, "Principles of object perception," *Cognitive Science*, vol. 14, no. 1, pp. 29–56, 1990.

[73] K. Koffka, *Principles of Gestalt Psychology*. Routledge, 1st ed., 1935.

[74] E. S. Spelke, "Core knowledge," *American Psychologist*, vol. 55, no. 11, pp. 1233–1243, 2000.

[75] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[76] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[77] V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, "An introduction to deep reinforcement learning," *Foundations and Trends® in Machine Learning*, vol. 11, no. 3-4, pp. 219–354, 2018.

[78] P. Dollár and C. L. Zitnick, "Structured forests for fast edge detection," in *2013 IEEE International Conference on Computer Vision*, pp. 1841–1848, 2013.

[79] B. Schölkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio, "Toward causal representation learning," *Proceedings of the IEEE*, vol. 109, no. 5, pp. 612–634, 2021.

[80] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *CoRR*, vol. abs/1312.6114, 2013.

[81] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[82] A. Zlatintsi, P. Koutras, G. Evangelopoulos, N. Malandrakis, N. Efthymiou, K. Pastra, A. Potamianos, and P. Maragos, "Cognimuse: a multimodal video database annotated with saliency, events, semantics and emotion with application to summarization," *EURASIP Journal on Image and Video Processing*, vol. 2017, no. 1, p. 54, 2017.

[83] G. Evangelopoulos, A. Zlatintsi, A. Potamianos, P. Maragos, K. Rapantzikos, G. Skoumas, and Y. Avrithis, "Multimodal saliency and fusion for movie summarization based on aural, visual, and textual attention," *IEEE Transactions on Multimedia*, vol. 15, no. 7, pp. 1553–1568, 2013.

[84] P. Koutras and P. Maragos, "Susinet: See, understand and summarize it," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 809–819, 2019.

[85] A. Tsiami, P. Koutras, and P. Maragos, "Stavis: Spatio-temporal audiovisual saliency network," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4765–4775, 2020.

[86] G. E. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *CoRR*, vol. abs/1503.02531, 2015.

[87] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *International Conference on Learning Representations*, 2017.

[88] G. Aydemir, W. Xie, and F. Guney, "Self-supervised object-centric learning for videos," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[89] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, "Dream to control: Learning behaviors by latent imagination," in *International Conference on Learning Representations*, 2020.

[90] Z. Wu, N. Dvornik, K. Greff, T. Kipf, and A. Garg, "Slotformer: Unsupervised visual dynamics simulation with object-centric models," in *The Eleventh International Conference on Learning Representations*, 2023.

[91] J. Kenney, T. Buckley, and O. Brock, "Interactive segmentation for manipulation in unstructured environments," in *2009 IEEE International Conference on Robotics and Automation*, pp. 1377–1382, 2009.

[92] P. Fitzpatrick, "First contact: an active vision approach to segmentation," in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, vol. 3, pp. 2161–2166 vol.3, 2003.

[93] H. van Hoof, O. Kroemer, and J. Peters, "Probabilistic segmentation and targeted exploration of objects in cluttered environments," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1198–1209, 2014.

[94] L. J. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *CoRR*, vol. abs/1607.06450, 2016.

[95] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, 2012.

[96] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser, "Learning synergies between pushing and grasping with self-supervised deep reinforcement learning," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4238–4245, 2018.

[97] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (Y. Bengio and Y. LeCun, eds.), 2015.

[98] D. Hendrycks and K. Gimpel, "Bridging nonlinearities and stochastic regularizers with gaussian error linear units," *CoRR*, vol. abs/1606.08415, 2016.

[99] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 386–397, 2020.

[100] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, 2016.