

**POLITECNICO DI MILANO**  
Corso di Laurea Magistrale in Ingegneria Informatica  
Dipartimento di Elettronica e Informazione



**ODIN: UN SISTEMA COGNITIVO  
BIOISPIRATO PER  
L'APPRENDIMENTO  
SENSOMOTORIO**

**AI & R Lab**  
Laboratorio di Intelligenza Artificiale  
e Robotica del Politecnico di Milano

**Relatore: Prof.ssa Giuseppina Gini**  
**Correlatore: Dott. Ing. Flavio Mutti**  
**Correlatore: Prof. Riccardo Manzotti**

**Tesi di Laurea di:**  
**Francesco Ferrini, matricola 755508**  
**Francesco Gallo, matricola 755748**

**Anno Accademico 2012-2013**



*A chi c'è stato, sempre*



# Sommario

Nell'ambito della robotica si è sempre cercato di costruire degli agenti autonomi capaci di imitare le caratteristiche dell'essere umano. Partendo da architetture nelle quali il robot era costretto a comportarsi secondo regole rigide preimpostate al suo interno, si sono sviluppate nuove aree di ricerca basate sullo sviluppo cognitivo: la robotica bioispirata. Questa ha portato alla creazione di sistemi cognitivi capaci di evolversi ma ancora troppo semplici per poter collaborare con un sistema motorio bioispirato. Lo scopo della nostra tesi è di presentare una architettura bioispirata capace di imparare e gestire le informazioni sensoriali in modo da poter cooperare con un modello motorio bioispirato basato sul sistema composto da cervelletto, gangli della base e corteccia motoria. Abbiamo quindi costruito un modello motorio complesso dal quale abbiamo estrapolato le informazioni necessarie per lo sviluppo di una architettura cognitiva che abbiamo chiamato ODIN. I risultati ottenuti con gli esperimenti riguardanti i test di riconoscimento visivo e uditivo per il sistema proposto, hanno prodotto risultati molto incoraggianti per future sperimentazioni e integrazioni.



# Ringraziamenti

## Francesco

Il primo ringraziamento va a me stesso, per essere riuscito in qualcosa che a un certo punto non credevo più possibile.

Ringrazio Valeria, la mia sorellina, con la quale ho instaurato un rapporto veramente bello, ora forza Vale, la prossima dottoressa sarai te! Grazie sister!

Ringrazio la Marta, anche detta Mé, per tutto il sostegno ricevuto, la pazienza, la perseveranza, e il coraggio avuti nel sopportarmi in questi 6 anni e passa. Sei fantastica e unica!

Ringrazio i miei genitori, Etta e Luciano, per la loro pazienza e per il loro appoggio incondizionato. Siete veramente delle rocce!

Ringrazio Matteo perchè rende felice una delle tre donne più importanti della mia vita. Continua così e migliorati sempre! Ringrazio in generale tutta la mia famiglia che mi è sempre stata vicino nei momenti più difficili della mia vita. Senza di voi non sarei assolutamente mai arrivato fin qui!

Ringrazio Francesco perché questa tesi è di tutti e due e ce la meritiamo tantissimo entrambi. Allons-y Gillo! \*pugnetto\*

Ringrazio Vittorio&Romana, grazie per avermi accolto in famiglia e avermi fatto sfuggire ogni tanto al grigiore milanese!

Ringrazio la Professoressa Gini per avermi permesso di intraprendere il lavoro di tesi con Lei e per il supporto dato durante la stesura dell'elaborato. Grazie veramente!

Ringrazio Flavio per tutto l'aiuto ricevuto e l'entusiasmo mostrato durante il lavoro svolto. Sei un grande, ma mi devi ancora insegnare a schiacciare!

Ringrazio Riccardo per essere sempre stato entusiasta del lavoro che stavamo svolgendo. In bocca al lupo per l'esperinza all'MIT!

Ringrazio Antonella, grazie a lei ogni mattinata partiva sempre con il sorriso sulla bocca. Grazie mille Anto!

Ringrazio tutti gli amici milanesi, Simo per il suo essere troppo figo, Emil

perchè non sa guardare con gli occhi del serpente, Daniele perchè sta passando tanti esami e quindi farà una festa a casa sua, l'Ilaria perchè sopporta Daniele e non mi ha ancora insultato per la molestia dimostrata al suo matrimonio, Claudio perchè essere un Valderman con poteri illimitati non ha prezzo, Cesco perchè è sempre stato puntuale, Valentina per le avvincenti sfide tennistiche. Grazie veramente tanto anche a tutti voi!

Ringrazio tutta la Solese, a partire dal presidente Carlo fino al magazziniere Lino, passando per tutti i compagni di squadra che ci sono e che sono passati, Tommy, Tia Carminiello, Ivan, Cagno, Riky, Cecco il tronista, Flavio il bello, Davidone&Ale (la coppia di fatto!), Mirko, Pietro, Kharja (anche detto Hamza), il grandissimo capitano Palu, Perro, Faso, Paolone, Luca, il bell'addormentato Furlo, il mister Uccelli e mister Pava, il DS Pino, il fisioterapista Ciccio che mi ha rimesso in sensto più di una volta, il guardialinee Vito e Pini, per l'avventura passata nella stagione 2011/2012, me la ricorderò a vita ne sono certo!

Ringrazio Tommy & Tia anche per tutti i viaggi fatti insieme: Heads Will Roll! Ringrazio anche tutta l'Algo, il Polimi Calcio, Fiz, Sganx & Marco, Biase, Lore, Prema, Cesco, Luzzi, Lollo, e tutti gli altri, e l'Atletico PerNulla con il suo direttor. lup. mann. figl. di putt. Ing. Marco, perchè mi hanno fatto crescere come persona e come sportivo.

Ringrazio enormemente tutti gli amici delle Marche, Tony&Elisa (tanto vi sposerete prima di me), Daniele&Valentina (tanto vi sposerete prima di me), Adriano&Chica (tanto vi sposerete prima di me), Pax&Vale (vi siete già sposati :) ), Malato&MalatinaFromHell (purtroppo so che non vi sposerete prima di me), Rezzolo, Manuel, Paolò & il suo peproncino, Ebola, Federico & La Locanda Del Riccio, Eros, Carlo, Peppe&Elena, Cif e le sue figure da cioccolataio, il Capo per tutte le volte che viene a fare il pollo alle partite di poker e tutti quelli che in qualche modo mi sono stati vicini tra Spiaggetta e scopa ladra che purtroppo mi sono certamente dimenticato, perchè ogni volta che sono tornato nelle mie amate Marche l'affetto è rimasto sempre immutato. Siete dei veri amici!

Ringrazio Seby per tutti i suoi utilissimi consigli e soprattutto per tutte le volte che mi ha rimesso in sesto fisicamente. Grande accappatoio!!!

Ringrazio tutti i professori che mi hanno concesso l'onore di avere qualche 30 e 30L nella carriera universitaria! È sempre un piacere ammirarli per le ultime volte attraverso il webpoliself!

Ringrazio il Politecnico per l'istruzione ricevuta.



## **Francesco**

Ringrazio l'arciere per aver supportato la scelta di questa tesi estremamente impegnativa e per gli anni di cazzeggio passati insieme.

Un grazie alla custode delle chiavi dell'artefatto che balla, all'uomo dalle scarpe ninja e l'agendina e all'uomo di ferro che gioca a pallavolo e ha una statuetta in miniatura di sé stesso davanti al pc senza i quali questo lavoro non esisterebbe.

Un dovuto ringraziamento anche all'atleta, il disegnatore e lo snowboarder che ci hanno fatto trovare già lastricato l'inizio della strada e alla Gandalf del dipartimento che ha reso migliore ogni giorno passato al Poli.

Un grazie di cuore alla gigantessa dal multiforme ingegno e accento, alla sua grande pazienza e alle nostre lunghe parentesi aperte e mai chiuse.

A Mr SKazzato che è scappato in UK per non sentire più la Carrà , Bargnani trepassineldelirio Jr che continua ad organizzare partite di calcetto e basket, a LeBron Fante e i suoi orridi ramarrì marroni sperduti in qualche capitale mondiale, allo scienziato pazzo che prima o poi debellerà l'idiozia, a Marcelinho che riuscirà a mangiare un frutto del mare, all'uomo con la barba che sottosottosotto è un pochino ingegnere e al nostro Maradona Jr che è perennemente in ritardo ma che alla fine c'è quasi sempre.

Allo spartano nato nel secolo sbagliato che prima o poi diventerà il legittimo successore di Hokuto, alla ragazza dei controsterzi che oltre a sopportare il suo moroso sopporta anche me, alla piccola circense che fa un salame di cioccolato stellare, alla strana coppia che tralasciando la mania per queLgiocOonLine sono una bella coppia, al figlio illegittimo di Moscardelli, quello di Roque Junior e quello di Mesbah che sono i seguaci di Javalone e del divino Jonathan e alla fotografa che è riuscita nell'arduo compito di fotografare il mio possente stacco da terra.

Alla lettrice che-sognava-di-esser-trovata-su-una-spiaggia per i suoi meravigliosi sorrisi, all'angelo piangente che altrimenti mi picchia, a coleichenon-vuoleesserenominata che prima o poi mi farà mangiare la sua torta, alla regina che diventerà imperatrice del mondo, alla elettronica che nonostante tutto c'è ancora, all'uomo senza trolley che si è sorbito le mie paranoie, alla ragazza con il tenero peluche ma non al suo relatore, alla Chef che non ha

un profilo ma una galleria degna di un paio di stelline Michelin, all' arbitro canturino che sottosotto è un pistolero, al vero Adam Kadmon che ormai non è più noto con il suo nome di battesimo, alla temporeggiatrice paranoica che non è un orso come si pensa.

A Sfondaaaa che OUUUUU, al creatore di personaggi sfigati che deve ancora rispettare la promessa, a coluichenonpuòvincereasioscurosignore che ha avuto l'onere di lavorare con me nella tesi precedente, a capitano Ovvio che guarda con gli occhi del serpente la salsiccia sulla brace, alla ragazza dall'orecchio di perla che ha domato il tamarro, allo slavo-saraceno che è un clone migliore dell'originale, alla centaura che ha domato 1 cane 2 gatti e un vampiro arabo, al burino che litiga con ogni allenatore e alla maestrina che mi ha rifocillato di ottimo cibo ed è più in forma del marito, a Tinky Winky che come presidente dell'Atletico Pernulla non fa niente da mattina a sera.

Alla Trilli castana che inizia a pensare che non sia poi tanto normale, a mr Google che a breve vedremo sulla copertina del Time, al pistoiese che continua ad invitarmi a vedere la sua bella moglie e la splendida figlia, all'uomo della Magnum che se non sfonderà nel campo della musica sarà il modello più bello in modo assurdo, a colei che possiede il potere del bio che ci ha accolti in casa molto meglio del fratello, ai giocatori professionisti di scopa ladra, beachvolley o semplici scaldasedie che mi ha fatto passare una favolosa settimana di vacanza, alla pianista che si è proposta per la lettura e correzione della tesi, alla designer con il ghigno dipinto e il suo particolarissimo stile creepy, ai quattro scumbinati meno uno e al mondo che gli gira attorno.

Alla mia famiglia.

A tutti quelli che mi hanno deluso, facendomi crescere e a tutti quelli che mi sono involontariamente dimenticato.

A me stesso, perché in fondo me lo merito.

È stato un lungo e difficile viaggio ma con voi è stato indubbiamente piacevole. Questo percorso è finito e, a breve, spero di poterne iniziare un altro, sempre con il vostro supporto.

Quindi..... ALLONS-Y!

## Francesco

Ringrazio Franz per aver supportato la scelta di questa tesi estremamente impegnativa e per gli anni di cazzeggio passati insieme.

Un grazie alla professoressa Gini, a Riccardo e Flavio senza i quali questo lavoro non esisterebbe.

Un dovuto ringraziamento anche a Luca, Marco e Alessio che ci hanno fatto trovare già lastricato l'inizio della strada e ad Antonella che ha reso migliore ogni giorno passato al Poli.

Un grazie di cuore a Camilla, alla sua grande pazienza e alle nostre lunghe parentesi aperte e mai chiuse.

A Skivo che è scappato in UK per non sentire più la Carrà, Steve che continua a organizzare partite di calcetto e basket, Simo e i suoi orridi amari maroni spediti in qualche capitale mondiale, a Dag che prima o poi debellerà l'idiozia, a Dave che riuscirà a mangiare un frutto del mare, a Kora che sottosottosotto è un pochino ingegnere e a Bobo che è perennemente in ritardo ma che alla fine c'è quasi sempre.

A Luca che prima o poi diventerà il legittimo successore di Hokuto, a Gai che oltre a sopportare il suo moroso sopporta anche me, a Bea che fa un salame di cioccolato stellare, a Bruno e Stefania che traslocando la mania per due giochi online sono una bella coppia, a Rondi, Mosa e Basso che sono i segretari di Javahone e del divino Jonathan e ad Alice che è riuscita nell'arduo compito di fotografare il mio possente stacco da terra.

A Caterina per i sui meravigliosi sorrisi, a Marta che altrimenti mi picchia, a Chiara che prima o poi mi farà mangiare la sua torta, a Giulia che diventerà imperatrice del mondo, a Federica che nonostante tutto c'è ancora, a Marco che si è sorbito le mie paranoie, ad Anna ma non al suo relatore, a Valeria che non ha un profilo ma una galleria degna di un paio di stelle Michelin, a Gherro che sottosotto è un pistolero, al Cosch che ormai non è più noto con il suo nome di battesimo, a Maddalena che non è un orso come si pensa.

A Sino che OUUUUU, a Robi che deve ancora rispettare la promessa, a Gigi che ha avuto l'onore di lavorare con me nella tesi precedente, a Emil che guarda con gli occhi del serpente la salsa sulla brace, a Elisa che ha domato il tamaro, a Claudio che è un clone migliore dell'originale, a Micol che ha domato i cani, 2 gatti e un vampiro arabo, a Daniele che litiga con ogni allenatore e Lara che mi ha ricolitato di ottimo cibo ed è più in forma del marito, a Marco che come presidente dell'Atletico Pernilla non fa niente da mattina a sera.

A Terry che inizia a pensare che non sia poi tanto normale, a Stef- no che a breve vedremo sulla copertina del Time, a Kara che continua ad invitarmi a vedere la sua bella moglie e la splendida figlia, a Morris che se non stonderà nel campo della musica sarà il modello più bellollobel- loimmodosarudo, ad Annalisa che ci ha ascoltati in casa molto meglio del fratello, alla complicità marchigiana che mi ha fatto passare una favolosa settimana di vacanza, ad Alessandra che si è proposta per la lettura e correzione della tesi, a Elisa e al suo particolarissimo stile creedy, a Enzo, Giovanni, Toni e al mondo che gli gira attorno.

Alla mia famiglia.

A tutti quelli che mi hanno deluso, facendomi crescere e a tutti quel- li che mi sono involontariamente dimenticato.

A me stesso, perché in fondo me lo merito.

È stato un lungo e difficile viaggio ma con voi è stato indubbiamente piacevole. Questo percorso è finito, è a breve, spero di poterne iniziare un altro, sempre con il vostro supporto.  
Quindi..... ALLONS-Y!

# Indice

<b>Sommario</b>	<b>I</b>
<b>Ringraziamenti</b>	<b>III</b>
<b>1 Introduzione</b>	<b>1</b>
1.1 Inquadramento generale del problema . . . . .	1
1.2 Lavori correlati . . . . .	3
1.3 Scopo di questo lavoro . . . . .	4
1.4 Struttura della tesi . . . . .	5
<b>2 Stato dell'arte</b>	<b>7</b>
2.1 Il cervello . . . . .	7
2.1.1 Reazione agli stimoli . . . . .	7
2.1.2 Lo sviluppo cognitivo . . . . .	12
2.1.3 La generazione del movimento attraverso le primitive di moto . . . . .	13
2.1.4 Teorie sulla popolazione dei neuroni . . . . .	14
2.2 Apprendimento automatico . . . . .	16
2.2.1 Apprendimento non supervisionato, supervisionato e con rinforzo . . . . .	16
2.2.2 Classificatori One Versus All e binari . . . . .	17
2.2.3 Apprendimento Hebbiano e reti neurali . . . . .	21
<b>3 Impostazione del problema di ricerca</b>	<b>23</b>
3.1 Definizione del problema . . . . .	23
3.2 Come è stato affrontato il problema . . . . .	25
3.3 Scelte implementative . . . . .	26
<b>4 Architettura del sistema</b>	<b>29</b>
4.1 Filtraggio degli Ingressi . . . . .	29
4.2 Modulo Filogenetico . . . . .	30

4.3	Modulo Intenzionale . . . . .	30
4.3.1	Modulo di Categorizzazione . . . . .	32
4.3.2	Modulo Ontogenetico . . . . .	33
4.4	ODIN . . . . .	35
4.5	Il modello del sistema motorio . . . . .	39
<b>5</b>	<b>Implementazione</b>	<b>43</b>
5.1	Filtraggio degli Ingressi . . . . .	43
5.2	Modulo Filogenetico . . . . .	45
5.3	Modulo Intenzionale . . . . .	45
5.3.1	Modulo di Categorizzazione . . . . .	46
5.3.2	Modulo Ontogenetico . . . . .	48
5.4	ODIN . . . . .	49
<b>6</b>	<b>Risultati sperimentali</b>	<b>51</b>
6.1	Motivazioni . . . . .	51
6.2	Esperimento delle lettere . . . . .	53
6.2.1	Risultati . . . . .	53
6.3	Esperimento volti . . . . .	57
6.3.1	Risultati . . . . .	58
6.4	Esperimento audio . . . . .	59
6.4.1	Risultati . . . . .	60
<b>7</b>	<b>Conclusioni e sviluppi futuri</b>	<b>63</b>
7.1	Sviluppi Futuri . . . . .	65
	<b>Bibliografia</b>	<b>71</b>
<b>A</b>	<b>ICA</b>	<b>77</b>
A.1	Principi di funzionamento di ICA . . . . .	79
A.2	L'algoritmo . . . . .	82
A.3	Conclusioni . . . . .	84
<b>B</b>	<b>Esempi Dataset</b>	<b>85</b>

# Elenco delle figure

2.1	Aree di interesse del cervello. . . . .	8
2.2	Esempio di scomposizione degli stimoli visivi da parte del cervello umano: in particolare possiamo vedere i vari livelli in cui vengono applicati i filtri; tra questi si notano i filtri riguardanti il colore e il bianco e nero e quelli inerenti alla ricerca dei lineamenti. È possibile anche avere una visione d'insieme sulle vie che vengono utilizzate nella trasmissione delle informazioni. . . . .	9
2.3	Visione d'insieme delle vie centrali uditive che si estendono dai nuclei cocleari alla corteccia uditiva. L'immagine è molto particolareggiata ma vuole solo dare una visione generale della trasmissione dell'informazione uditiva [38]. . . . .	11
2.4	L'homunculus somatosensoriale rappresenta l'organizzazione somatotopica delle afferenze somatosensitive del corpo. Alcune regioni del copro (e.g. mano e bocca) sono visivamente più grandi di altre a causa di una maggiore presenza di recettori cutanei in esse presenti. . . . .	12
2.5	Combinazione delle primitive motorie. . . . .	14
2.6	Modello del sistema Actor-Critic così come introdotto nell'articolo <i>Reinforcement Learning: Introduction</i> [62]. . . . .	17
2.7	Esempio di curva ROC con valutazione del test. . . . .	19
2.8	Un esempio di classificazione di un classificatore binario: l'area tratteggiata contiene gli elementi che sono stati riconosciuti come appartenenti alla stessa classe ( $TP + FN$ ), mentre l'area in blu contiene tutti gli elementi che sono realmente corretti. . . . .	20

2.9	Esempi di problemi linearmente separabili e non: nelle figure (a) e (b) si può notare come il piano sia diviso in due sezioni, quella contenente il segno “+” e quella contenente il segno “-”; nella figura (c) sarebbe impossibile con una sola linea separare le due differenti figure perché il problema è non linearmente risolvibile. . . . .	22
4.1	Modulo Intenzionale. . . . .	32
4.2	Modulo di Categorizzazione. . . . .	33
4.3	Modulo Ontogenetico. . . . .	34
4.4	Architettura ODIN senza la propagazione dei segnali ontogenetici. . . . .	37
4.5	Architettura ODIN con la propagazione dei segnali ontogenetici. . . . .	38
4.6	Aree di apprendimento nel cervello come presentato da Kenji Doya [21, 22]. . . . .	40
4.7	Modello del sistema motorio. . . . .	42
5.1	Visuale a cono della rappresentazione HSV. . . . .	44
5.2	Esempio di immagine a cui è stato applicato il filtro di Canny. . . . .	44
5.3	Esempio di segnale filtrato tramite FFT. . . . .	45
5.4	Esempio di Moduli Intenzionali con più ingressi. . . . .	47
6.1	Esempio di diverse K . . . . .	53
6.2	Risultati del Segnale Rilevante del classificatore della lettera A. Le prime 160 immagini corrispondono a quelle che il riconoscitore dovrebbe classificare correttamente mentre le successive 1440 sono di altre lettere. . . . .	54
6.3	Esempio di foto dello stesso soggetto illuminate diversamente. . . . .	57
6.4	Esempi di immagini dal database A e le corrispondenti del database B. . . . .	57
6.5	Esempio di foto di ogni singolo soggetto utilizzato per l’esperimento. . . . .	58
6.6	Esempio di immagini scartate. . . . .	58
6.7	Risultati del test audio; i campioni che dovrebbero essere riconosciuti vanno dal 51 al 100. . . . .	60
6.8	Curva ROC dei campioni audio. . . . .	61
7.1	Un esempio di possibile modellizzazione della parte motoria associata a più sistemi ODIN. . . . .	67



7.2	Possibile modello che implementi il riconoscimento delle parole. I vari moduli ODIN riconoscono la singola lettera (come specificato nella figura) e un sistema di classificazione successivo, che potrebbe anch'esso essere costituito da un nuovo sistema ODIN, impara le parole riconosciute. . . . .	69
A.1	ICA (a) segnali originali; (b) segnali mixati; (c) componenti indipendenti; (d) segnali ricostruiti. . . . .	78
A.2	Esempio di distribuzione gaussiana . . . . .	81
B.1	Esempio di immagini di addestramento per la singola lettera.	85
B.2	Esempio di immagini di test per la singola lettera. . . . .	86
B.3	Esempio di immagini di test per un singolo soggetto. . . . .	86
B.4	Esempio di immagini di addestramento per un singolo soggetto.	87
B.5	Esempio di immagini presenti nel database, in particolare quelle appartenenti al primo soggetto preso in considerazione.	88



# Elenco delle tabelle

6.1	Risultati del test con le lettere. . . . .	56
6.2	Precision, accuracy e recall del sistema con l'analisi delle lettere.	56
6.3	Risultati del test audio. . . . .	60
6.4	Accuracy, precision e recall del test audio. . . . .	61



# Capitolo 1

## Introduzione

*“But actually from a non-linear, non-subjective viewpoint it’s more like a big ball of wibbly wobbly... timey wimey... stuff.”*

The Doctor, Doctor Who

Il problema della creazione di un agente intelligente, in grado cioè di saper agire e pensare come un essere umano, è materia di discussione sia di neuroscienziati che di ingegneri. Anche se questo problema non è di recente individuazione, solo negli ultimi tempi sono nati approcci, detti bioispirati, basati sulla biologia dell’essere umano. Il metodo utilizzato per proporre questi nuovi modelli si basa sull’analisi delle funzioni dell’essere umano, e il tentativo di riprodurne il comportamento con strumenti ingegneristici. In questo lavoro presentiamo una architettura bioispirata capace di imparare e gestire le informazioni sensoriali in modo da renderle utilizzabili per un modello motorio basato sul sistema cervelletto-gangli della base-corteccia motoria.

### 1.1 Inquadramento generale del problema

La robotica ha sviluppato e continua a sviluppare sistemi “*behaviour-based*” capaci di adattare il proprio comportamento ai cambiamenti del mondo in cui si trova, al fine di raggiungere i propri obiettivi. In questo approccio gli scopi sono preimpostati nel robot, senza che questi sia in grado di generarne di nuovi. La robotica epigenetica, invece, studia lo sviluppo cognitivo nei sistemi naturali e artificiali allo scopo di adattarsi all’ambiente e sviluppare autonomamente nuove motivazioni ed obiettivi che non erano previsti in fase di progettazione.

Per realizzare robot in grado di interagire in modo efficace con gli esseri umani e fare in modo che possano essere integrati nella loro vita quotidiana, bisogna interessarsi in particolar modo alla comprensione, dal punto di vista neuroscientifico, del cervello umano in modo da poter riprodurre i comportamenti tipici delle persone in un agente autonomo. In particolar modo bisogna fare attenzione a come il cervello umano percepisce ed interagisce con il mondo circostante; infatti, apprendere e gestire in maniera corretta gli input sensoriali è di fondamentale importanza per qualsiasi attività che un essere umano voglia compiere, soprattutto per la gestione del movimento.

Anche l'esecuzione di un compito particolarmente semplice, come quello di raggiungere la maniglia di una porta, richiede la disponibilità delle informazioni visive necessarie per rappresentare internamente una mappa con la posizione della maniglia nello spazio. Questo semplice atto necessita anche di informazioni propriocettive grazie alle quali è possibile ricostruire una rappresentazione interna del corpo in modo tale che al braccio possano essere inviati i comandi motori corretti. Le azioni volte al perseguimento di un obiettivo possono essere eseguite solo grazie al fatto che, le regioni del sistema nervoso centrale che controllano il movimento, hanno accesso al flusso di informazioni che continuamente raggiunge il sistema nervoso centrale. L'azione integrativa del sistema nervoso, cioè la decisione di eseguire un particolare movimento piuttosto che un altro, dipende perciò dall'interazione fra i sistemi motori e i sistemi sensoriali [38].

Per creare un robot bioispirato in grado di sviluppare le sue capacità intellettive e di gestire le informazioni motorie, così come farebbe un essere umano, è necessaria la modellizzazione delle aree cognitive e motorie del cervello. Le prime sono state presentate e successivamente implementate in una architettura chiamata IDRA [44, 9], mentre per le vie motorie non è mai stato presentato un sistema che comprendesse tutte le aree del cervello interessate (cervelletto, gangli della base e corteccia motoria). Risulta quindi particolarmente interessante creare un sistema interamente bioispirato che implementi sia la capacità cognitiva che quella motoria. Tuttavia l'architettura IDRA deficitava di una componente fondamentale: la gestione delle informazioni sensoriali.

La cooperazione tra un'architettura cognitiva e il modo in cui un agente percepisce e interagisce con l'ambiente in cui si trova, può portare verso la creazione di una nuova generazione di sistemi robotici autonomi.

## 1.2 Lavori correlati

Molte intelligenze artificiali e i loro rispettivi robot, sono sviluppati solamente per il raggiungimento di uno o più obiettivi specifici. Alcuni robot sono programmati con determinate abilità che permettono l'interazione con l'ambiente circostante e con l'uomo. Molti di questi effettuano una selezione dei loro comportamenti preesistenti che gli permette di reagire, in accordo con le passate esperienze, in modo da raggiungere l'obiettivo fissato a priori [5]. Questo tipo di robot viene comunemente chiamato "*behaviour-based*".

Mentre i robot *behaviour-based* esplorano l'ambiente per ottimizzare le loro azioni per raggiungere obiettivi predefiniti, un'altra tipologia di robot è in grado di esplorare l'ambiente circostante per "trovare" nuovi obiettivi da perseguire. Si tratta dei "*motivational-based*" [44] e si differenzia dai primi proprio per la capacità di compiere azioni seguendo degli obiettivi che non sono stati impostati durante la programmazione della loro intelligenza.

Il sistema IDRA si è concentrato sull'implementazione dell'architettura cognitiva tentando solamente nell'ultima fase di sperimentazione di unirli ad una parte motoria realizzata ad hoc per l'esperimento. Questo esperimento consiste nel far muovere al robot un solo braccio nella cui mano tiene un oggetto colorato, in modo che catturi il suo interesse, e cercando di fargli imparare autonomamente quali azioni eseguire al fine di massimizzare il suo soddisfacimento, cioè al fine di osservare l'oggetto. Se la parte cognitiva risulta bioispirata e con buoni risultati nel primo test proposto riguardante il riconoscimento delle forme, la stessa cosa non si può dire del suo collegamento con il sistema motorio e il relativo esperimento. La parte cognitiva si basa su un Modulo Intenzionale capace di astrarre concetti (e.g. imparare approssimativamente la forma di un oggetto ritenuto interessante), utile in situazioni semplici ma piuttosto slegata dalla parte motoria; infatti, durante l'esecuzione di movimenti pseudocasuali cercava di memorizzare, in una matrice stato-azione, tutte le coppie che portassero il robot in una posizione che gli permettesse di avere un alto valore di soddisfazione.

L'idea sulla quale si basa l'esperimento è valida, ma nel corso della pianificazione del test non sono stati tenuti in considerazione due aspetti fondamentali: la componente bioispirata del movimento e il collegamento tra la parte cognitiva e quella motoria. La componente bioispirata risulta fondamentale nella corretta progettazione dei movimenti da compiere e nell'individuare le caratteristiche necessarie per la corretta rappresentazione del mondo in cui l'agente si deve muovere. I sensi servono, nello specifico, per migliorare l'esperienza dell'agente nello spazio in cui può interagire e senza di essi non è possibile ottenere risultati soddisfacenti nel tentativo di emu-

lare il comportamento umano. Il primo passo per la pianificazione di un movimento si basa sulla rappresentazione del mondo esterno che il cervello effettua grazie all'apparato sensoriale.

Tuttavia, bisogna dire che con IDRA si voleva dimostrare la validità del sistema cognitivo proposto e l'esperimento motorio era stato elaborato solamente per cercare di capire le potenzialità di questa architettura.

### 1.3 Scopo di questo lavoro

Dopo aver studiato l'implementazione di IDRA, il loro modello cognitivo e aver capito cosa serviva per poter creare un modello che unisse le due architetture, cognitiva e motoria, abbiamo deciso di creare il sistema proposto in questa tesi.

Ci siamo proposti, quindi, di creare un sistema sufficientemente potente in grado di riuscire nella gestione di diversi tipi di segnali sensoriali. Abbiamo sfruttato l'idea e la realizzazione dell'architettura cognitiva già presente in IDRA per creare un nuovo sistema più complesso, funzionale a una successiva aggiunta di un sistema motorio bioispirato.

Per realizzare questo tipo di architettura abbiamo creato un modello motorio basato su cervelletto, gangli della base e corteccia motoria, studiando il funzionamento neuroscientifico del cervello umano. La creazione di questo modello ci ha permesso di studiare i requisiti necessari per il suo corretto funzionamento, da implementare in una architettura bioispirata, capace di interpretare e gestire i segnali provenienti dai sistemi sensoriali. Per mantenere il collegamento con l'architettura IDRA, nome di un mostro mitologico, e allo stesso tempo differenziarci da essa, abbiamo deciso di chiamare il nostro sistema ODIN, poiché nella mitologia norrena Odino era anche il dio della conoscenza, rimanendo sempre in ambito mitologico ma cambiando pantheon.

In questa tesi porteremo tre esperimenti a supporto della validità della nostra architettura: due riguardanti la gestione dei segnali visivi e uno relativo ai segnali audio.

Nel primo test l'architettura viene testata attraverso il riconoscimento di immagini di lettere. Questo esperimento ha validità sia dal punto di vista biologico, infatti l'essere umano si cimenta continuamente con ottimi risultati nella lettura, sia da un punto di vista ingegneristico, poiché il problema del riconoscimento dei caratteri appartiene ad un campo di ricerca ormai consolidato.

Nel secondo esperimento ci siamo concentrati, invece, sul riconoscimento dei volti sempre per le due tipologie di motivi visti in precedenza: da



un punto di vista biologico il riconoscimento dei volti è una delle capacità che possono essere riscontrate fin dall'infanzia, mentre, da un punto di vista ingegneristico, risulta interessante provare il sistema con immagini più complesse rispetto alle precedenti.

Con il terzo esperimento, quello riguardante i segnali audio, vogliamo verificare che il sistema sia in grado di gestire un segnale differente da quello visivo. Questo esperimento ha lo scopo di mostrare le potenzialità dell'architettura nella gestione di segnali provenienti dagli apparati sensoriali che sono di supporto a quello visivo durante il movimento.

I risultati ottenuti, valutati con diversi metodi, supportano la validità dell'architettura.

## 1.4 Struttura della tesi

**Nel Capitolo 2** illustriamo quali teorie sono alla base della nostra tesi, dimostrando come le conoscenze scientifiche e ingegneristiche cooperino tra loro.

**Nel Capitolo 3** presentiamo i lavori precedentemente svolti nello stesso ambito e le motivazioni che hanno portato alla realizzazione di questa tesi.

**Nel Capitolo 4** descriviamo il sistema che abbiamo creato (ODIN) da un punto di vista architetturale, utilizzando un approccio dal basso verso l'alto, mettendo in risalto le similitudini tra il sistema proposto e il cervello.

**Nel Capitolo 5** presentiamo le tecniche implementative utilizzate per la realizzazione di ODIN, utilizzando la stessa struttura del Capitolo 4.

**Nel Capitolo 6** descriviamo gli esperimenti svolti per verificare la bontà del sistema ODIN.

**Nel Capitolo 7** analizziamo i risultati degli esperimenti precedentemente mostrati e indichiamo i possibili sviluppi futuri.

**Nell'Appendice A** presentiamo una descrizione teorica dell'algoritmo utilizzato per l'Analisi delle Componenti Indipendenti (*Independent Component Analysis*, ICA).

**Nell'Appendice B** mostriamo alcuni esempi dei dataset utilizzati per i nostri esperimenti.



## Capitolo 2

# Stato dell'arte

*“Nothing shocks me. I’m a scientist.”*

Dr. Henry Walton Indiana Jones, Jr., *Indiana Jones and the Temple of Doom*

Il lavoro che presentiamo in questa tesi è basato su conoscenze informatiche utilizzate per modellizzare ed implementare strutture neuroscientifiche. In questo capitolo presentiamo brevemente i concetti base su cui si fondano i nostri modelli: una unità di apprendimento generico e un modello bioispirato per il controllo dei movimenti. Partiremo con una descrizione del funzionamento biologico del cervello umano soffermandoci sulla risposta agli stimoli, per poi analizzare alcuni modelli teorici e matematici creati per emularne il comportamento. Tratteremo anche i modelli utilizzati in ambito di apprendimento automatico e i classificatori in grado di stimarne la bontà.

### 2.1 Il cervello

Il cervello umano è composto da diverse parti, alcune delle quali ancora poco note. Per i nostri scopi, quelle di interesse sono: il talamo, l'amigdala, la corteccia, i gangli della base e il cervelletto. Anche se ognuna di queste parti svolge più compiti in diverse aree di interesse, noi ci siamo soffermati sulle interazioni presenti in ambito motorio, nella selezione dell'azione da compiere e nella gestione di stimoli ritenuti interessanti.

#### 2.1.1 Reazione agli stimoli

Il cervello umano è composto da circa cento miliardi di cellule nervose e lo studio delle principali connessioni tra queste cellule, soprattutto dal punto

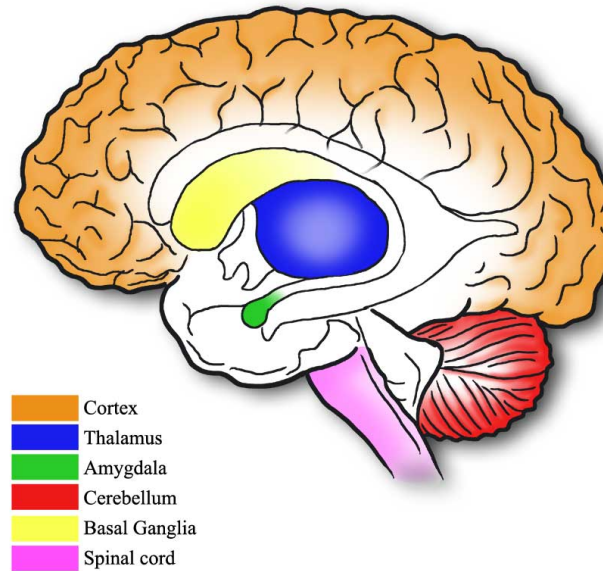


Figura 2.1: Aree di interesse del cervello.

di vista funzionale, ha dato un apporto fondamentale alla comprensione dei processi che sono alla base dell'attività mentale.

Le modalità di percezione differenti, quali la visione, il tatto o l'udito, vengono elaborate in maniera simile da sistemi sensoriali differenti. I recettori sensoriali scompongono, in prima analisi, uno stimolo nelle sue componenti elementari. Questi recettori reagiscono a differenti tipologie di stimolazione e, una volta attivati, rispondono con un particolare tipo di scarica rappresentante un particolare aspetto dello stimolo. Queste informazioni vengono ritrasmesse attraverso collegamenti formati da una serie di cellule fino a specifiche aree della corteccia celebrale. Le regioni corticali in corrispondenza delle quali sono rappresentate modalità sensoriali differenti, comunicano tramite collegamenti intracorticali con delle regioni associative multimodali, cioè quelle regioni che selezionano e integrano i segnali che ricevono dalle differenti regioni unimodali.

Le ricerche sulla visione, ad esempio, hanno dimostrato che le informazioni visive arrivano dalla retina al cervello attraverso almeno due vie parallele distinte: la via *M* e la via *P*. La separazione delle informazioni visive inizia nella retina grazie a due diverse tipologie di popolazione di cellule gangliari costituite, rispettivamente, da cellule grandi (cellule *M*) e cellule piccole (cellule *P*). Queste due tipologie cellulari portano informazioni alquanto differenti a diversi strati del talamo: le cellule *M* proiettano

agli strati magnocellulari del corpo genicolato laterale del talamo (via M), mentre quelle P proiettano agli strati parvicellulari (via P) [38]. Ciascuna delle vie viene utilizzata per l'analisi di aspetti differenti della stessa immagine visiva (e.g. forma, colore, etc) e queste informazioni, trasmesse per mezzo di vie funzionali distinte, vengono integrate nella corteccia per creare una rappresentazione dell'immagine nel cervello, coerente con quella percepita, attraverso un insieme di regole intrinseche nel sistema nervoso centrale. Una semplice rappresentazione di quanto appena descritto è visibile nella Figura 2.2

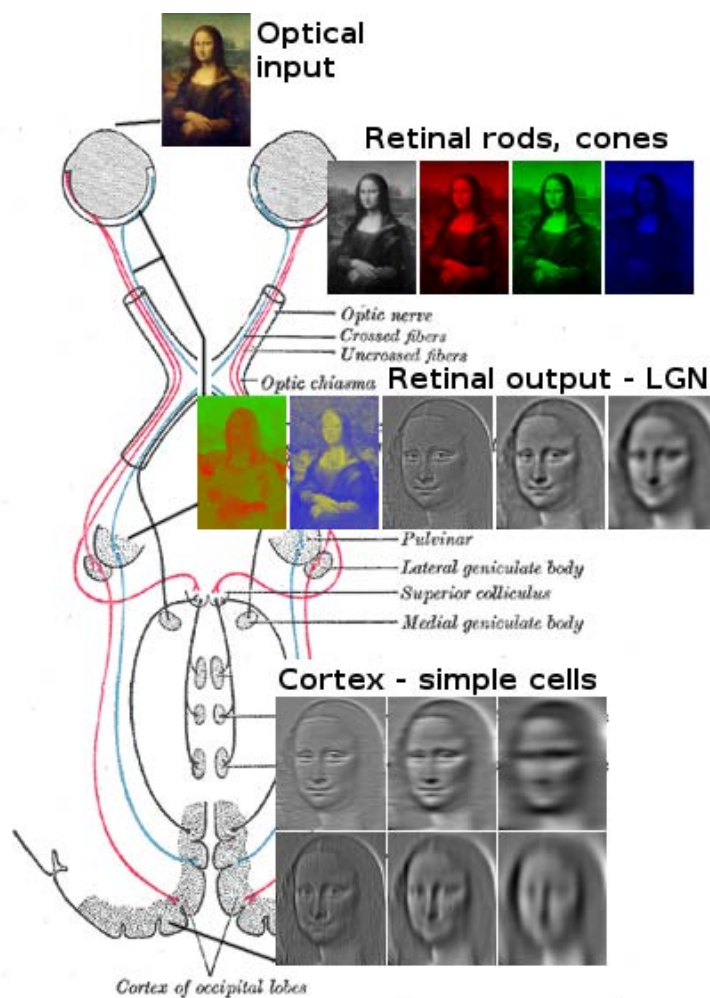


Figura 2.2: Esempio di scomposizione degli stimoli visivi da parte del cervello umano: in particolare possiamo vedere i vari livelli in cui vengono applicati i filtri; tra questi si notano i filtri riguardanti il colore e il bianco e nero e quelli inerenti alla ricerca dei lineamenti. È possibile anche avere una visione d'insieme sulle vie che vengono utilizzate nella trasmissione delle informazioni.

L'udito è un senso fondamentale per le comunicazioni umane e l'elaborazione del suo segnale con la raccolta dei suoni da parte dell'orecchio. Un insieme molto grande di cellule ciliate identifica le singole frequenze che compongono lo stimolo e le codifica tramite la frequenza di scarica delle fibre del nervo acustico [38]. Le informazioni acustiche vengono, quindi, trasmesse dalle cellule ciliate ai nuclei cocleari le cui fibre codificano sia la frequenza che l'intensità di un suono. Queste informazioni, visto che le vie uditive centrali si estendono dai nuclei cocleari fino alla corteccia uditiva, vengono quindi inviate alle aree preposte all'analisi dei segnali uditivi della corteccia cerebrale. Una visione d'insieme delle vie uditive è visibile in Figura 2.3.

I sistemi sensoriali dell'uomo, tra cui udito e vista, formano rappresentazioni interne del corpo e del mondo esterno. Una delle principali funzioni di queste ultime è quella di aiutare e guidare il movimento. Anche il semplice gesto di afferrare una maniglia necessita di almeno due tipi di informazioni: quelle visive riguardanti la posizione della maniglia, allo scopo di costruire una rappresentazione grafica, e quelle propriocettive per mezzo delle quali è possibile costruire una rappresentazione interna del corpo, in modo tale da trasmettere i comandi motori appropriati. Le azioni di questo tipo, mirate cioè all'esecuzione di un movimento per il raggiungimento di uno scopo specifico, possono essere eseguite grazie al fatto che le regioni preposte al controllo del movimento facenti parte del sistema nervoso centrale hanno accesso al flusso di informazioni generali che continuamente lo raggiungono. Questa azione decisionale del sistema nervoso dipende quindi dall'interazione tra il sistema motorio e i sistemi sensoriali.

Esistono molteplici vie, anatomicamente distinte, che proiettano in parallelo dai centri motori fino alle aree adibite all'analisi dei segnali motori della corteccia cerebrale, passando per il midollo spinale, per il tronco dell'encefalo fino al cervelletto e ai gangli della base. Come accade per i sistemi sensoriali (si veda la Figura 2.4), la maggior parte delle aree motorie della corteccia cerebrale è organizzata in maniera somatotopica; infatti, i movimenti di parti adiacenti del corpo sono controllate da aree contigue del sistema nervoso centrale [38].

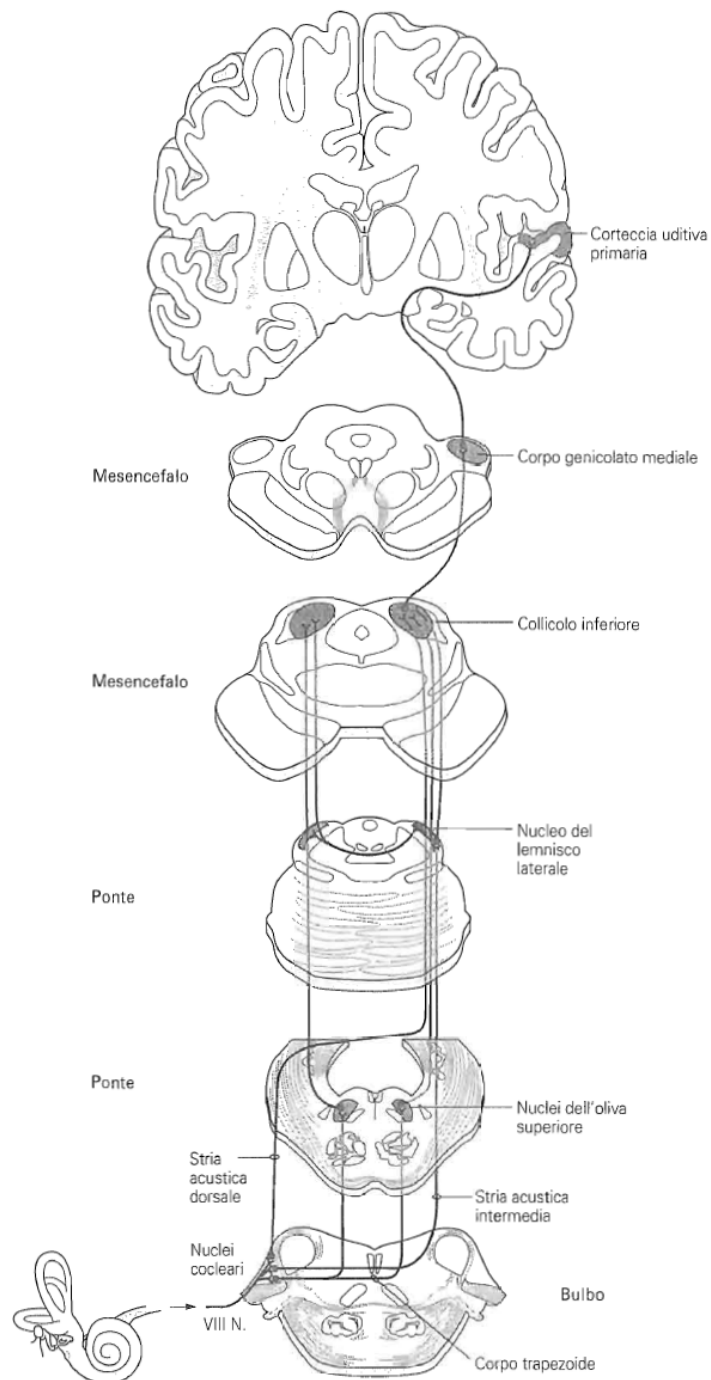


Figura 2.3: Visione d'insieme delle vie centrali uditive che si estendono dai nuclei cocleari alla corteccia uditiva. L'immagine è molto particolareggiata ma vuole solo dare una visione generale della trasmissione dell'informazione uditiva [38].

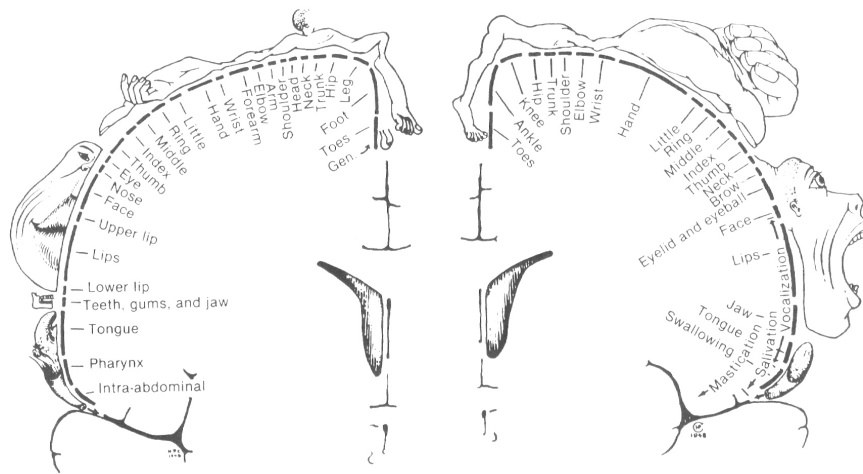


Figura 2.4: L'homunculus somatosensoriale rappresenta l'organizzazione somatotopica delle afferenze somatosensitive del corpo. Alcune regioni del corpo (e.g. mano e bocca) sono visivamente più grandi di altre a causa di una maggiore presenza di recettori cutanei in esse presenti.

### 2.1.2 Lo sviluppo cognitivo

A partire dalle informazioni di base sulla trasformazione, trasmissione ed elaborazione degli stimoli a cui il corpo umano è soggetto, sono stati compiuti molti studi riguardanti il campo neuroscientifico e psicologico sullo sviluppo cognitivo. Questi studi si focalizzano sullo studio dello sviluppo di un bambino in termini di elaborazione dell'informazione (e.g. comprensione e apprendimento del linguaggio). In altre parole, lo sviluppo cognitivo è l'abilità di pensare e capire [59].

L'importanza del modello amigdala-talamo-corteccia nello sviluppo cognitivo è stato mostrato da diversi studi [38, 61]. L'amigdala è un piccolo corpo nervoso a forma di mandorla il cui compito è la valutazione dell'aspetto emotivo degli stimoli ricevuti basandosi sugli istinti innati ed esperienze pregresse. È il responsabile della gestione delle paure e delle altre emozioni. Il talamo, invece, serve a filtrare e instradare le informazioni oltre a sviluppare nuove motivazioni ed obiettivi. È direttamente collegato con la corteccia di cui è una "mappa in miniatura" [67]. Infine, la corteccia cerebrale è la parte più esterna del cervello; è suddivisa in diversi settori (corteccia parietale posteriore, corteccia premotoria, corteccia motoria, etc) e riceve segnali dagli organi sensoriali; ha anche il compito di immagazzinare ricordi, elaborare pensieri e gestire il linguaggio.



A partire da queste conoscenze è stato compiuto un lavoro di tesi [9] che ha portato alla creazione di un modello di apprendimento chiamato *Intentional Module* che permette, partendo da uno stimolo associato ad un piacere innato come la saturazione di un oggetto, l'apprendimento di nuovi obiettivi tramite ciò che viene osservato dal sistema (e.g. imparare ad apprezzare una forma a partire dal fatto che l'oggetto in questione sia colorato).

Vedremo in seguito, nel Capitolo 3, come questo modello sia servito da base all'architettura che proponiamo.

### 2.1.3 La generazione del movimento attraverso le primitive di moto

La capacità del cervello umano di creare al suo interno una rappresentazione del proprio corpo e dell'ambiente circostante permette l'invio, da parte del cervello stesso, di comandi motori appropriati al movimento che si vuole svolgere. La corteccia, dopo aver ricevuto le informazioni provenienti dall'apparato sensoriale, pianifica l'azione che deve essere compiuta e invia la decisione presa ai gangli basali e al cervelletto; i primi si occupano di controllare la traiettoria durante lo svolgimento dell'azione, mentre il secondo valuta se la posizione raggiunta corrisponde con quella attesa e, nel caso in cui non lo sia, segnala la presenza di un errore di esecuzione. Gli impulsi da inviare alla corteccia dopo il movimento vengono trasmessi attraverso il talamo, mentre tutte le informazioni che devono essere mandate ai muscoli per eseguire il movimento vengono spedite attraverso il midollo spinale.

Corteccia, gangli della base e cervelletto sono stati modellizzati in letteratura tramite differenti tipi di apprendimento presentati in seguito. La corteccia agisce tramite un comportamento non supervisionato mentre i gangli secondo un modello con rinforzo [23, 63] e in particolare con un modello actor-critic [8] in cui i gangli della base svolgono la parte dell'actor mentre la parte di critic è affidata al cervelletto [22, 21, 17]. Quest'ultimo è modellizzato attraverso uno schema supervisionato.

Il midollo spinale è la più bassa parte caudale del sistema nervoso; riceve e processa le informazioni sensoriali da varie parti del corpo e controlla il movimento dei muscoli e delle articolazioni [38]. Funge da ponte tra la mente e il corpo, ricevendo impulsi elettrici dal sistema nervoso periferico e mandando questi segnali al sistema nervoso centrale. Allo stesso modo, riceve impulsi dal cervelletto, dalla corteccia e anche dai gangli della base processandoli per mandare istruzioni corrette a specifici muscoli.

Alcuni studi hanno illustrato come i movimenti complessi siano generati da una combinazione limitata di segnali a forma d'onda, indipendenti

rispetto al muscolo considerato; questi pochi segnali controllano l'attività dei muscoli in modo da produrre cambiamenti cinematici [35]. Questi studi suggeriscono che il sistema nervoso non abbia bisogno di generare tutti gli schemi di attività di tutti i muscoli del corpo, ma che possa crearne solamente alcuni generici per poi combinarli tra loro per generare l'attivazione di uno specifico muscolo attraverso la modulazione della fase e dell'ampiezza di queste primitive [51]. Muscoli differenti svolgono azioni differenti mostrando cambiamenti diversi negli schemi di attivazione. Questo modello può essere rappresentato da un oscillatore che produce una frequenza in uscita basata sul segnale di ingresso, composto da una funzione pesata per la generazione dello schema d'attivazione del muscolo e da un insieme di funzioni non lineari che modulano l'uscita dell'oscillatore [52].

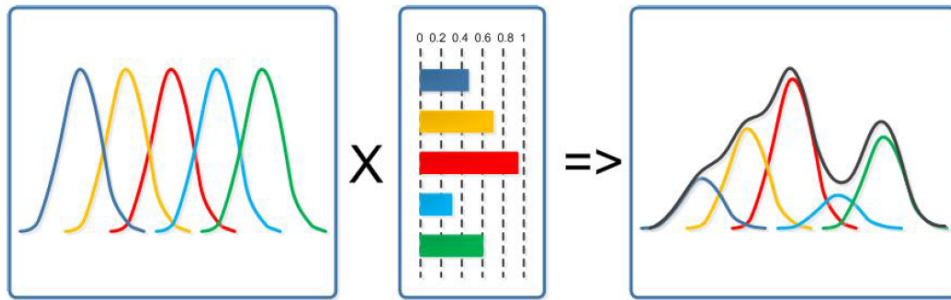


Figura 2.5: Combinazione delle primitive motorie.

Le primitive motorie sono modellizzate come una specifica rete neurale che si trova nel midollo spinale e che genera il movimento definendo il segnale di attivazione per i muscoli [28]. Ogni primitiva motoria è uno schema di attivazione di base con differenti pesi associati ad ogni muscolo. Studi effettuati in questo campo hanno mostrato come la formulazione delle primitive motorie attraverso equazioni differenziali non lineari sia stata applicata con successo in molte applicazioni robotiche e possa essere impiegata con gli algoritmi di apprendimento supervisionato e di rinforzo per l'ottimizzazione delle traiettorie [58].

Il nostro lavoro non ci porterà nell'analisi approfondita delle primitive motorie ma useremo comunque questo concetto e queste teorie per la costruzione di un modello motorio completo e implementabile, come vedremo nei capitoli successivi.

#### 2.1.4 Teorie sulla popolazione dei neuroni

Anche se la maggior parte dei settori della corteccia riceve segnali da sorgenti specifiche, è stato provato che la stessa area risponde prontamente anche

a nuovi stimoli [20]. Quindi, tutta la corteccia è composta dallo stesso tipo di cellule ed è in grado di rispondere, immagazzinare e adattarsi a diversi tipi di segnali. Il cervello umano è in grado di risolvere problemi computazionalmente differenti grazie ad un grande numero di aree funzionali. Anche se questo numero elevato di aree specifiche lascia pensare che sia presente un meccanismo indipendente per ogni problema, si è visto che alla base di tutte queste aree si trovano dei meccanismi computazionali comuni. Questi meccanismi si sviluppano a partire da una parte computazionale comune che sottostà ogni area funzionale: la rete neurale. Due di questi meccanismi sono la *normalizzazione* e il *population coding* [56, 4].

Il primo consiste nel mediare la somma delle risposte delle attività della popolazione di neuroni, dividendo tale valore per il numero di neuroni coinvolti, allo scopo di codificare una rappresentazione neurale. L'idea della normalizzazione nasce per giustificare la risposta fisiologica dei neuroni nella corteccia primaria visiva e, in seguito, in altre aree del cervello; questo ha portato a sostenere che la normalizzazione sia comune nelle reti neurali[11]. Viene utilizzato anche al di fuori del cervello, come nella retina.

Il population coding, invece, consiste nel codificare le informazioni sensoriali attraverso un gruppo di neuroni. Ogni neurone risponde a determinati stimoli e la risposta del gruppo osservato può essere analizzata per trovare nuove informazioni riguardo a quel segnale iniziale. Questi neuroni, infatti, sono spesso organizzati in strati o regioni così che neuroni vicini abbiano risposte simili[26]. Sembra che, oltre alla codifica dei movimenti, il population coding intervenga anche nel processo decisionale[19], nel movimento oculare per focalizzare lo sguardo (movimento saccadico) e nel riconoscimento di schemi visivi [7].

Queste teorie, tuttavia, mostrano la necessità di studiare ulteriormente il funzionamento dei gruppi di neuroni poiché sembrerebbe possibile modellizzarne il funzionamento tramite formule statistiche ancora in fase embrionale [7]. Una teoria che prende spunto dal population coding è quella secondo la quale l'essere umano ha una capacità notevole nel riconoscimento di schemi. Ad esempio si sostiene che Garry Kasparov, campione mondiale di scacchi, abbia imparato circa 100,000 posizioni delle pedine sulla scacchiera, o che Shakespeare abbia utilizzato per i suoi componimenti circa 100,000 differenti concetti (utilizzando le stesse parole con diversi significati), sottolineando come un esperto in un particolare ambito arrivi a conoscere fino a 100,000 differenti sfumature di conoscenza nel suo campo. Viene sottolineato anche come ci siano mezzo milione di colonne corticali nella neocorteccia umana (la parte del cervello che è, in termini di evoluzione, più recente ed unica dei mammiferi) e che ogni riconoscitore di schema all'interno di una colonna

corticale contenga circa 100 neuroni. Questo significa che esistono più di 300 milioni di riconoscitori di schemi all'interno della neocorteccia [39].

Alcune di queste teorie avrebbero bisogno di raccogliere più evidenze scientifiche a supporto delle loro tesi, sia da un punto di vista neurologico che da un punto di vista matematico. In questo lavoro viene presentato un modello riguardante la popolazione dei neuroni che prende spunto dal concetto di base che lega tutte le teorie appena descritte senza implementarne uno specifico, come sarà possibile vedere nei capitoli che seguiranno.

## 2.2 Apprendimento automatico

Nell'ambito dell'intelligenza artificiale uno dei campi più importanti è quello dell'apprendimento automatico che consiste nel fare imparare al sistema un modo per risolvere dei compiti autonomamente. L'algoritmo usato, modificando dei parametri interni chiamati pesi, fa in modo che l'agente riesca ad adattarsi al compito da svolgere.

### 2.2.1 Apprendimento non supervisionato, supervisionato e con rinforzo

Esistono tre principali tipi di algoritmi di apprendimento automatico: con rinforzo, non supervisionato e supervisionato.

Quando il sistema deve compiere delle azioni in un ambiente (e.g. raggiungere un obiettivo in una mappa) si utilizza l'*apprendimento per rinforzo*. L'apprendimento per rinforzo è una tecnica di apprendimento di nuovi comportamenti basato su un segnale di rinforzo; ogni azione svolta riceverà una valutazione, detta rinforzo, che ha lo scopo di giudicare se il comportamento del sistema sia stato corretto o meno. Lo scopo dell'agente è quello di portare a termine il proprio compito andando così a massimizzare la valutazione finale. L'agente può essere un animale, un essere umano o un sistema artificiale, come ad esempio un robot o un programma per computer, mentre il segnale di rinforzo può essere rappresentato da cibo, acqua, soldi o qualsiasi cosa possa misurare le performance dell'agente [23].

La tecnica di apprendimento per rinforzo viene utilizzata da tempo con successo in ambito di controllo ottimo. Uno dei più importanti approcci di questa tipologia di apprendimento riguarda l'uso delle tecniche classiche di controllo ottimo per addestrare e supervisionare un sistema attraverso ripetute interazioni con l'ambiente [37]. Si possono trovare esperimenti svolti in ambiti reali che dimostrano come sia possibile ottenere buoni risultati mediante l'uso di un approccio di apprendimento per rinforzo come, ad

esempio, il controllo del volo di un elicottero autonomo acrobatico [2] oppure la capacità di superare gli ostacoli del robot quadrupede *LittleDog* [27]. Di particolare interesse è il modello actor-critic in cui la valutazione viene fatta da un modulo dedicato, il critico [62].

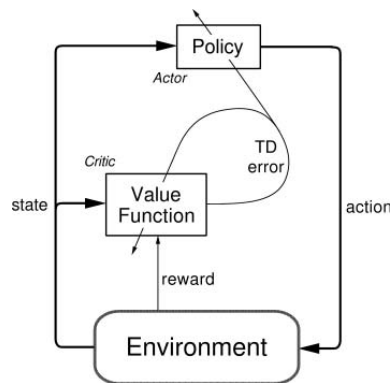


Figura 2.6: Modello del sistema Actor-Critic così come introdotto nell'articolo Reinforcement Learning: Introduction [62].

L'apprendimento per rinforzo, che è stato sviluppato attraverso intuizioni sulla teoria di apprendimento degli animali, sembra dare una spiegazione coerente al funzionamento dei gangli della base [22, 21, 23, 63].

L'*apprendimento non supervisionato* prevede il passaggio di dati al sistema senza fornire ulteriori informazioni come uscite previste o ricompense in base alle azioni svolte. Lo scopo è di riuscire a trovare degli schemi tra i dati e riuscire a raggruppare i differenti gruppi focalizzandosi sulla classificazione.

Quando invece, oltre ai dati di ingresso, vengono forniti anche i valori che ci si aspetta in uscita dal sistema si parla di *apprendimento supervisionato*. In questo modo il sistema impara a produrre i risultati attesi a partire dai dati in ingresso.

Mentre l'approccio *mixed model*, basato cioè sull'utilizzo di differenti modelli cooperanti, è molto ben conosciuto dalla comunità neuroscientifica, la sua applicazione nel controllo dei robot non è ancora molto diffusa [18].

### 2.2.2 Classificatori One Versus All e binari

La classificazione è il problema di identificare a quale categoria appartenga una nuova osservazione sulla base di addestramento dei dati contenenti osservazioni la cui categoria è nota. L'algoritmo che implementa concretamente la classificazione è detto classificatore. Di seguito presentiamo due tipologie di algoritmi: classificatori binari e *One Versus All*.

Il classificatore binario si occupa di dividere un dato insieme di elementi in due diversi gruppi: quelli che appartengono a una classe  $X$  e quelli esclusi. Questa semplice tipologia di compito ha diverse applicazioni pratiche che spaziano dal campo medico (riconoscere un paziente sano da uno malato), all'information retrieval (capire quali informazioni sono importanti rispetto all'interrogazione effettuata dall'utente) e, come avremo modo di spiegare meglio nei capitoli successivi, riconoscere uno specifico segnale discriminando tutti quelli differenti.

Questi classificatori necessitano di una soglia che permetta di scartare tutti gli elementi che si trovano al di sotto di questa e di includere quelli che la superano. La scelta di tale soglia non è sempre facile e spesso dipende dai dati analizzati oltre che dal compito richiesto [24]. Infatti, quando non si è già selezionato un valore a priori, si preferisce impostare una soglia di discriminazione pari a 0.5 anche se questa risulta molto scadente in presenza di un gran numero di valori molto alti o molto bassi. Si decide, quindi, di utilizzare degli indicatori di bontà del test per la selezione della soglia: *accuracy*, *precision*, *recall* e *specificity*.

L'accuracy, o accuratezza, rappresenta la percentuale di classificazioni corrette (positive e negative) rispetto al campione totale dei dati:

$$\frac{TP + TN}{TP + FP + TN + FN} \quad (2.1)$$

dove  $TP$  rappresenta i veri positivi (True Positive),  $TN$  indica i veri negativi (True Negative),  $FP$  si riferisce ai falsi positivi (False Positive) e  $FN$  denota i falsi negativi (False Negative).

La precision, o precisione, indica la percentuale di classificazioni positive e corrette ( $TP$ ) rispetto a tutte le classificazioni ritenute positive, comprendenti anche i falsi positivi:

$$\frac{TP}{TP + FP} \quad (2.2)$$

Il recall, o richiamo, rappresenta la percentuale di elementi correttamente riconosciuti del testing set rispetto a tutti gli elementi che sarebbero dovuti essere effettivamente classificati come veri positivi:

$$\frac{TP}{TP + FN} \quad (2.3)$$

La specificity, o specificità, indica la percentuale di elementi negativi reali correttamente classificati come tali:

$$\frac{TN}{TN + FP} \quad (2.4)$$

Un ulteriore strumento che possiamo usare per stimare la bontà dei risultati ottenuti dal test è la curva ROC (Receiver Operating Characteristic o Relative Operating Characteristic). Questa curva viene generata a partire da una funzione che lega la probabilità di ottenere un risultato vero positivo nella classe degli elementi da riconoscere (Recall) alla probabilità di ottenere un risultato falso positivo ( $1 - specificity$ ). L'area sottesa da questa curva (Area Under Curve, AUC) viene utilizzata per discriminare tra un test buono e uno insufficiente; si considera, infatti, l'area sottesa dalla curva come la percentuale di successo del test. Questo deriva dalla definizione dell'area stessa che viene utilizzata; la curva ROC, infatti, viene tracciata all'interno di un quadrato di area 1. Ne risulta, quindi, che un test perfetto coprirà completamente il quadrato avendo una percentuale di successo del 100%. Normalmente, per riconoscere il limite di ragionevolezza di un test, si confronta la curva ROC ottenuta con la bisettrice del primo quadrante (come mostrato in Figura 2.7): se la curva risulta inferiore a quest'ultima il test non ha prodotto risultati apprezzabili, avendo una probabilità di successo inferiore al 50%, infatti funzionerebbe meglio come classificatore una scelta casuale basata sul lancio di una moneta[25].

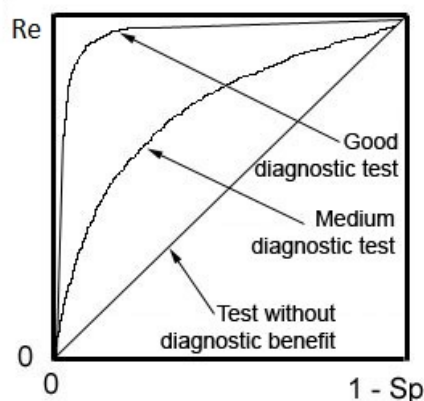


Figura 2.7: Esempio di curva ROC con valutazione del test.

Quando, invece, abbiamo a disposizione più gruppi in cui poter inserire un elemento si usano i classificatori multi classe. Il campione di addestramento consiste, quindi, in elementi appartenenti a  $N$  diverse classi e lo scopo è quello di costruire una funzione che, esaminando un nuovo elemento, sia in grado di predire correttamente a quale insieme questo appartenga.

Uno dei metodi più semplici per creare un classificatore multiclasse è quello di appoggiarsi ai classificatori binari appena illustrati; infatti basta

creare un numero sufficiente di diversi riconoscitori binari per eseguire la classificazione desiderata (e.g. supponendo di voler classificare le singole cifre, serviranno solo 10 classificatori binari, ognuno addestrato su una differente cifra). Durante l'esecuzione del test ogni riconoscitore assegna un valore al campione appena osservato, indicante la confidenza che l'elemento analizzato appartenga alla sua classe, (si veda la Figura 2.8).

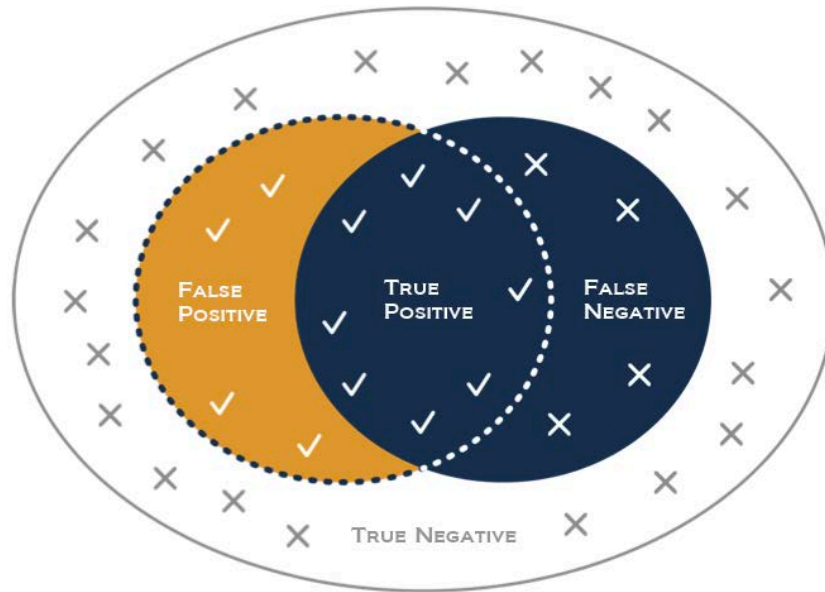


Figura 2.8: Un esempio di classificazione di un classificatore binario: l'area tratteggiata contiene gli elementi che sono stati riconosciuti come appartenenti alla stessa classe ( $TP + FN$ ), mentre l'area in blu contiene tutti gli elementi che sono realmente corretti.

Quando tutti i valori sono stati raccolti, si sceglie a quale classe appartiene l'elemento selezionando il classificatore che ha prodotto il valore più alto. La scelta di selezionare il valore più alto tra quelli assegnati dai singoli riconoscitori consente di aggirare la necessità del singolo classificatore binario di avere una soglia; infatti, non c'è nessun bisogno per il singolo classificatore di dividere nettamente i dati in due insiemi, poiché la procedura di classificazione viene fatta a posteriori, cioè una volta ottenuti tutti i valori di riconoscimento di ogni classificatore. Questo tipo di classificazione si chiama *One Versus All* (OVA). Pur essendo semplice ed intuitivo, sia computazionalmente che concettualmente, l' OVA permette di ottenere



risultati apprezzabili poiché simili, se non migliori, rispetto a metodi più sofisticati [54].

Oltre a quanto detto precedentemente, abbiamo deciso di utilizzare questo metodo di classificazione perché questo meccanismo risulta equivalente al *Winner-Take-All* (WTA) [36]. Questo meccanismo modella il funzionamento dei neuroni all'interno del cervello in determinate situazioni: quando i neuroni rispondono ad un determinato stimolo, per decidere quale risposta considerare, si seleziona quello con l'attivazione più intensa inibendo quelli con risposta minore [3]. L'OVA presenta la stessa tipologia di selezione andando a scegliere il valore più alto restituito dai classificatori, facendo così risultare nullo il contributo degli altri.

### 2.2.3 Apprendimento Hebbiano e reti neurali

Vi sono stati molti tentativi di produrre algoritmi di apprendimento biologicamente validi, ma il problema principale con cui tutte queste teorie si sono regolarmente imbattute è stato quello di permettere l'addestramento anche in presenza di funzioni non linearmente separabili. Tutte le tesi che hanno tentato di superare questo problema fanno capo alla regola di Hebb. La sua formulazione originale, del 1949, recita così: "Quando un assone della cellula A (la cellula presinaptica) prende parte ripetutamente nel processo di eccitamento della cellula B (la cellula postsinaptica), qualche cambiamento strutturale o metabolico subentra in una o entrambe le cellule in modo che l'efficienza di A, come cellula eccitatrice di B, aumenti" [29]. La più semplice delle formulazioni matematiche proposte per descrivere la regola di Hebb è la seguente:

$$\Delta w_{ij} = \eta x_i x_j \quad (2.5)$$

dove  $\Delta w_{ij}$  rappresenta la variazione dell'efficienza sinaptica tra il neurone  $i$  e il neurone  $j$ ,  $\eta$  la velocità di adattamento, mentre  $x_i$  e  $x_j$  rappresentano le cellule A e B.

Venne così proposto il perceptrone, un primo modello di neurone capace, grazie ai pesi assegnabili, di apprendere; questo modello, però, è in grado di risolvere solamente, come già anticipato, funzioni linearmente separabili. È stato mostrato come il perceptrone fosse completamente inutile nella risoluzione di problemi non linearmente separabili, come ad esempio il problema dello XOR [41]. Il perceptrone era in grado di separare un piano tramite un confine decisionale in due aree distinte, ma quando erano necessarie aree non linearmente separabili, come accade per lo XOR, il problema diventava irrisolvibile (vedere Figura 2.9).

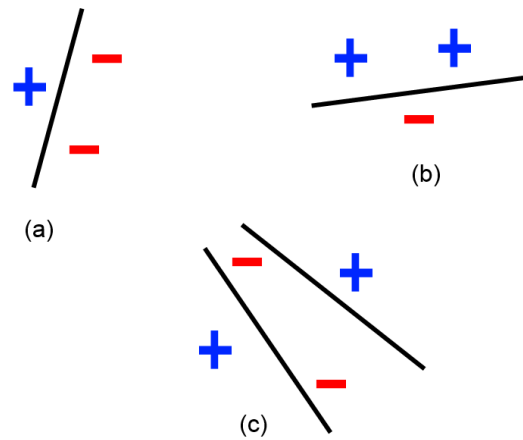


Figura 2.9: Esempi di problemi linearmente separabili e non: nelle figure (a) e (b) si può notare come il piano sia diviso in due sezioni, quella contenente il segno “+” e quella contenente il segno “-”; nella figura (c) sarebbe impossibile con una sola linea separare le due differenti figure perché il problema è non linearmente risolvibile.

Così, grazie all'introduzione delle reti neurali (un modello matematico che rappresenta l'interconnessione tra elementi definiti neuroni artificiali) composte da più layer di percettroni connessi tra loro, è diventato possibile risolvere anche i problemi non linearmente separabili.

Sia l'apprendimento hebbiano che le reti neurali vengono già utilizzati in ambito di robotica bioispirata per ricreare il funzionamento di alcune aree del cervello con risultati apprezzabili [46].

## Capitolo 3

# Impostazione del problema di ricerca

*“Certainty of death, \*small\* chance of success... What are we waiting for?”*

Gimli, The Lord of the Rings

In questo capitolo spieghiamo i motivi che ci hanno portato a compiere questo lavoro di tesi. Mostriamo, quindi, il lavoro dal quale siamo partiti e le ragioni che ci hanno portato a reingegnerizzare e ampliare quanto ottenuto precedentemente.

### 3.1 Definizione del problema

L'uomo, durante tutta la vita, migliora le proprie capacità mentali; questo processo è chiamato sviluppo cognitivo e si concentra su come una persona percepisce, pensa e prende coscienza del proprio mondo attraverso l'interazione con esso. Un aspetto fondamentale dello sviluppo cognitivo è la capacità di generare autonomamente nuovi obiettivi e comportamenti che permettono all'individuo di adattarsi a differenti situazioni. La tesi precedentemente proposta, riguardante questi aspetti, si è concentrata sulla costruzione di un modello robotico bioispirato, basato cioè sui processi del cervello umano, che permette ad un agente di sviluppare autonomamente nuovi obiettivi e comportamenti rispetto a quelli decisi a priori, cioè già presenti all'interno dell'architettura. Il suddetto lavoro di tesi, a sua volta, ha preso spunto da un articolo che ha modellizzato i moduli base per passare da un robot “behaviour-based” (guidato da comportamenti scelti a priori) a uno “motivations-based” (guidato dall'apprendimento di nuovi obiettivi) [44]. È stato quindi costruito un sistema, chiamato IDRA, che ricrea il

funzionamento delle aree del cervello che svolgono un ruolo fondamentale nello sviluppo cognitivo: amigdala, talamo e corteccia cerebrale.

Questo sistema si basa sull'Architettura Intenzionale, una rete di componenti elementari, chiamati Moduli Intenzionali. Ognuno di questi moduli contiene al proprio interno altri due moduli più specifici: il Modulo di Categorizzazione e il Modulo Ontogenetico. Il primo, che nel sistema emula la corteccia cerebrale, serve a mostrare l'attivazione neurale della corteccia in risposta allo stato del sistema stesso. Il Modulo Ontogenetico, invece, è il fulcro dello sviluppo di nuovi obiettivi e riproduce il funzionamento del talamo; riceve la risposta neurale dal Modulo di Categorizzazione e, tramite una funziona di apprendimento, sviluppa nuovi obiettivi. Quando termina di analizzare lo stato corrente, rilascia un segnale che indica quanto lo stato osservato rispecchi l'interesse verso i nuovi obiettivi. In aggiunta ai Moduli Intenzionali nella rete è presente anche un Modulo Filogenetico. Basato sull'amigdala, questo modulo contiene le "paure e gli istinti innati" dell'agente che, ovviamente, in questo caso sono state impostate a priori durante l'implementazione. Il segnale che viene generato e propagato dal Modulo Filogenetico rappresenta quanto lo stato attuale soddisfi gli obiettivi già fissati: quanto più lo stato è conforme a questi obiettivi, tanto più sarà alto il segnale. Il Modulo Intenzionale, dopo averli ricevuti dai moduli Ontogenetico e Filogenetico, analizza tali segnali e propaga quello più intenso. Oltre a questo valore, chiamato Segnale Rilevante, il Modulo Intenzionale propaga anche i valori ottenuti dal Modulo di Categorizzazione. Questi valori, nell'implementazione proposta, forniscono le distanze tra lo stato appena osservato e quelli memorizzati come nuovi obiettivi e, nel caso di Moduli Intenzionali connessi tra loro, questi dati vengono utilizzati come base per l'apprendimento del secondo modulo.

Nella tesi presa in considerazione, vengono anche svolti due esperimenti: il primo mostra l'apprendimento di una specifica forma mediante osservazione e il suo seguente riconoscimento in un ambiente differente, mentre il secondo vuole provare le capacità motorie dettate dal raggiungimento di un obiettivo prefissato. Per il primo esperimento si sono avuti dei buoni risultati per quanto riguarda l'apprendimento di nuovi obiettivi, mentre nel secondo, che voleva essere solo un tentativo iniziale di apprendimento del moto, sono emersi alcuni difetti architetturali; infatti, riproducendo gli esperimenti proposti, abbiamo notato che i movimenti del robot risultavano innaturali.

Analizzando l'architettura dal punto di vista biologico, ci siamo accorti che mancava la modellizzazione e, quindi, l'implementazione delle aree del cervello preposte alla gestione del movimento. Uno degli sviluppi futuri, infatti, riguardava la necessità di migliorare la parte motoria, per la qua-

le era stato proposto un modello non completo, comprendente una matrice stato-azione, che teneva traccia di alcuni movimenti importanti effettuati dal robot in fase di test, al posto dei modelli del cervelletto, della corteccia motoria e dei gangli della base. Una seconda osservazione riguarda il passaggio dei dati tra i diversi Moduli Intenzionali che doveva rappresentare idealmente il passaggio di informazioni attraverso la rete di neuroni del cervello umano. Abbiamo notato come, da un layer dell'architettura al successivo, il passaggio del vettore composto dalla concatenazione delle distanze del punto proiettato dai centroidi dei cluster creati di ogni Modulo Intenzionale afferente, fosse povero di informazioni rilevanti: più si creavano nuovi layer connessi ai precedenti, più i Moduli di Categorizzazione avevano problemi ad effettuare il clustering delle informazioni ricevute fino a non riuscire più a categorizzare alcunché. Oltre ad aver osservato che il vettore delle distanze del punto proiettato dai centroidi dei cluster contenesse effettivamente poche informazioni utilizzabili per la modellizzazione del passaggio di segnali tra i neuroni, abbiamo anche notato come la concatenazione delle informazioni non avesse basi biologiche. Infatti, come abbiamo visto nel capitolo precedente, nel nostro cervello, una volta arrivate le informazioni scomposte dagli organi sensoriali, viene eseguita una procedura di unificazione delle informazioni per la comprensione degli stimoli a cui si è soggetti.

Abbiamo quindi creato un modello motorio bioispirato, basato cioè sul cervelletto, sui gangli della base, sulla corteccia motoria e la loro interazione, al fine di studiarne e analizzarne gli elementi cognitivi necessari al suo funzionamento. Questi elementi ci hanno guidato nella creazione di un modello cognitivo bioispirato in grado di gestire tutte le informazioni necessarie per supportare la parte motoria.

## 3.2 Come è stato affrontato il problema

La creazione di un modello motorio basato sul cervelletto, sui gangli della base e sulla corteccia motoria risulta particolarmente stimolante poiché anche in letteratura non sono presenti molti sistemi che uniscano le varie aree del cervello preposte al controllo motorio del corpo umano [21]. Analizzando i risultati di IDRA abbiamo scoperto come tale sistema non si adatti alla gestione delle informazioni sensoriali necessarie per la parte motoria; anche l'esperimento mostrato presenta delle lacune, sia da un punto di vista del movimento eseguito, sia da un punto di vista biologico. Abbiamo quindi deciso di implementare un nostro sistema cognitivo bioispirato in grado di gestire i segnali sensomotori, partendo da quanto di buono dimostrato in IDRA.

Analizzando attraverso letteratura neuroscientifica i collegamenti presenti tra l'apparato sensoriale e il sistema motorio, abbiamo creato un modello motorio bioispirato che comprendesse le aree del cervelletto, dei gangli della base e della corteccia motoria. Questo modello si basa su un sistema cognitivo in grado di processare le informazioni provenienti dall'ambiente in cui l'agente deve potersi muovere.

Abbiamo bisogno quindi di un sistema in grado di imparare, riconoscere e gestire le informazioni provenienti dall'apparato sensoriale, in modo da supportare, in maniera adeguata, un modello motorio bioispirato. IDRA, utilizzando un singolo modulo intenzionale per senso, non era in grado di fornire informazioni valide al sistema motorio. Il nostro sistema cognitivo si pone come obiettivo proprio l'analisi e il passaggio di informazioni sensoriali adeguate al modello motorio.

Abbiamo creato una architettura su più layer basata sull'idea del Modulo Intenzionale proposta già in letteratura [44] e utilizzata anche nell'architettura IDRA [9]. La prima problematica affrontata è stata quella del passaggio di informazioni da un layer al successivo. In particolare, abbiamo apportato un cambiamento strutturale che prevede l'eliminazione del vettore passato dal Modulo di Categorizzazione ai moduli del layer successivo e, quindi, la scelta di un diverso segnale in ingresso al Modulo Intenzionale. Come anticipato nel paragrafo precedente, questa scelta si è resa necessaria per la scarsa rilevanza delle informazioni trasmesse con il metodo originale. Abbiamo quindi analizzato, dal punto di vista biologico, le possibili implicazioni riguardanti la comunicazione tra neuroni vicini (si veda il population coding presentato nel Capitolo 2). In seguito ad alcune ricerche effettuate, abbiamo deciso di utilizzare lo stesso segnale prodotto dai filtri come ingresso per tutti i Moduli Intenzionali. In particolare si è scelto di propagare in ingresso ad ogni Modulo Intenzionale i segnali in entrata di tutti i Moduli Intenzionali del layer precedente ad esso collegati, secondo lo schema che verrà presentato nel Capitolo 4; nel caso in cui siano presenti più moduli che convergono in un unico Modulo Intenzionale, il suo segnale in ingresso non sarà, come è previsto dall'architettura IDRA, la concatenazione dei segnali precedenti, ma la loro combinazione. Infatti, come visto in precedenza, i segnali delle differenti regioni associative unimodali vengono integrati tra loro nelle regioni associative multimodali del cervello [38].

### 3.3 Scelte implementative

Il nostro lavoro di tesi è stato svolto totalmente in MATLAB (*Matrix Laboratory*), un ambiente per il calcolo numerico e l'analisi statistica, che

comprende anche l'omonimo linguaggio di programmazione creato dalla *MathWorks*. La scelta dell'ambiente di sviluppo è stata dettata dalla praticità con la quale, attraverso questo strumento, è possibile riprodurre e modificare il lavoro di tesi precedente al nostro. Una volta analizzato il codice scritto nel linguaggio di programmazione *C#*, abbiamo riprodotto, tramite l'utilizzo di funzioni matematiche per l'analisi di matrici e vettori, l'intera tesi. In questa tesi, a differenza del lavoro precedente, non sono stati eseguiti lavori su di una macchina robot. Per testare il nostro sistema non ci siamo avvalsi dell'ausilio del robot antropomorfo NAO (come invece proposto nella tesi precedente) poiché ci siamo concentrati maggiormente sulla validazione del modello matematico piuttosto che sulla sua possibile implementazione in un agente autonomo.

Poiché lo scopo è di verificare la potenzialità del sistema, l'uso di un robot avrebbe introdotto delle non idealità che avrebbero potuto influire sui risultati (e.g. problemi di acquisizione delle immagini dovute alle condizioni di luce o riflessi su eventuali fogli), è per questo motivo che abbiamo preferito rimandare agli sviluppi futuri questa ulteriore fase di sperimentazione.

Una delle possibili e più interessanti espansioni di questo lavoro di tesi sarebbe, infatti, quella di riuscire ad implementare il sistema motorio nella sua totalità (unendo la parte di apprendimento con la parte inerente al movimento) e provare il sistema così ottenuto su un sistema autonomo antropomorfo. Il robot preso in considerazione dalla tesi precedente sarebbe un ottimo punto di partenza, ma sicuramente si rivelerebbe troppo semplice dal punto di vista strutturale, rispetto all'implementazione di tutte le caratteristiche insite nel moto umano.

Per quanto riguarda la validazione del modello, per essere ragionevolmente sicuri che la nostra architettura possa essere sufficientemente performante da poter imparare e riprodurre i movimenti (gestiti tramite primitive motorie [35]), abbiamo sviluppato tre test atti a verificarne l'efficacia, che riguardano input sensoriali di due diverse tipologie: visivi e uditivi. I primi due sono stati pianificati sin dall'inizio per poter fornire, oltre a un'analisi delle prestazioni, anche supporto al modello motorio, fornendo informazioni riguardanti l'ambiente circostante. L'ultimo viene proposto per mostrare le potenzialità dell'architettura in relazione a tipologie di segnali differenti da quelli video. In particolare, i test visivi si differenziano per la complessità delle immagini analizzate; nel primo caso ci soffermiamo sul riconoscimento di alcune lettere alfabetiche stampate, un compito ormai poco complesso e con prestazioni superiori al 97%[30, 64], mentre nel secondo esperimento utilizziamo immagini più complesse come quelle dei volti umani. Per entrambi i test ci siamo avvalsi dell'ausilio di dataset appropriati, disponibili

gratuitamente tramite internet.

Il dataset delle lettere contiene i 26 caratteri dell'alfabeto inglese dei quali ne abbiamo utilizzati 10. Ogni lettera è rappresentata da 800 esempi differenti sia per cambiamenti di font che per uso di corsivo e grassetto. I volti provengono dal database di Yale[1] e sono foto di diversi soggetti in differenti condizioni di luminosità. Il database è diviso in due gruppi, A e B: il primo è composto da foto che vanno dal collo ai capelli e che presentano anche elementi dello sfondo, mentre il secondo comprende solamente i volti, visualizzando unicamente la parte dal mento alla fronte. Per il nostro esperimento, come mostreremo nel Capitolo 6, abbiamo utilizzato il gruppo B. Infine, per l'esperimento audio, abbiamo utilizzato alcune tracce presenti su un sito internet, utilizzabili a scopo di ricerca[49]. A causa del metodo catalogazione e per la tipologia delle registrazioni stesse, abbiamo dovuto apportare alcune modifiche per fare in modo che fossero esaminate correttamente dal nostro sistema, come spiegato sempre nel Capitolo 6.



## Capitolo 4

# Architettura del sistema

*“We are ka-tet. We are one from many.”*

Roland Deschain, *The Dark Tower*

In questo Capitolo mostriamo l’architettura che abbiamo sviluppato e il suo funzionamento senza addentrarci nei particolari implementativi che verranno discussi nel Capitolo successivo. L’approccio utilizzato è dal basso verso l’alto, iniziando dal funzionamento delle parti atomiche dell’unità base dell’architettura, il Modulo Intenzionale, arrivando così alla spiegazione del comportamento dello stesso; per concludere illustriamo la rete completa che è stata creata per l’apprendimento generico.

### 4.1 Filtraggio degli Ingressi

Prima di spiegare il funzionamento dei vari moduli dobbiamo illustrare brevemente ciò che sta alla base del sistema: i filtri. Qualsiasi stimolo esterno (sia questo visivo, acustico o tattile) viene filtrato e rielaborato dal cervello, infatti, è stato provato che il cervello scompone le immagini prima di rielaborarle. Esistono due vie parallele, distinte anche dal punto di vista fisico oltre che funzionale, la via M e la via P. Queste vie si occupano di analizzare aspetti differenti della stessa immagine: la prima si occupa di filtrare l’immagine concentrandosi sul contrasto di luminanza, al fine di trovare i contorni degli oggetti presenti, mentre la seconda si focalizza sui contrasti, applicando dei filtri (e.g. bianco e nero, blu, rosso e verde). Anche l’apparato uditivo presenta un insieme di filtri per analizzare i suoni. Questi, come visto in letteratura, permettono di codificare sia l’intensità che la frequenza dei suoni [38].

Per questo motivo ai segnali in ingresso, a prescindere dalla loro tipologia, vengono applicati dei filtri che mirano a ricreare il sistema di filtraggio del cervello umano. Quando in ingresso al sistema si presenta un segnale visivo, per simulare il funzionamento delle vie M e P, applichiamo due filtri che permettono di imitarne il comportamento. Il primo trasforma l'immagine da colorata in scala di grigi, mentre il secondo, usando un algoritmo di ricerca dei contorni (*edge detection*), trasforma l'immagine mostrandone solamente i bordi. Quando in ingresso al sistema si presenta invece un segnale audio, viene applicato un filtro, come spiegato in precedenza, in grado di estrarre l'ampiezza del segnale.

## 4.2 Modulo Filogenetico

Prima di illustrare il funzionamento del modulo intenzionale, dobbiamo presentare il modulo filogenetico. Il suo ruolo, dal punto di vista biologico, corrisponde a quello dell'amigdala che, dopo aver ricevuto informazioni dall'apparato sensoriale, valuta l'interesse della situazione in cui l'organismo si trova in base a quello che è il comportamento innato[45, 40]. L'istinto è l'inclinazione naturale degli esseri viventi ad assumere comportamenti che non sono frutto di apprendimento o scelte consapevoli (e.g. gli animali che fuggono dal fuoco o i bambini che tendono naturalmente a preferire gli oggetti colorati rispetto a quelli in bianco e nero [14]).

Per riprodurre il funzionamento dell'amigdala nel sistema abbiamo creato delle funzioni istintive predefinite all'interno del Modulo Filogenetico.

Questo modulo ha come ingresso le informazioni che arrivano dall'apparato sensoriale, ognuna delle quali viene poi analizzata dalle funzioni istintive relative al senso di appartenenza (e.g. ciò che viene osservato è processato solo dalle funzioni istintive relative alla vista).

L'uscita del Modulo Filogenetico è un segnale che indica quanto ciò che viene percepito sia importante per il sistema in base agli istinti predeterminati.

## 4.3 Modulo Intenzionale

Il Modulo Intenzionale, che è l'unità fondamentale della nostra architettura, ha la capacità di apprendere e acquisire interesse verso nuovi stimoli; tale caratteristica, in neuroscienze, è chiamata neuroplasticità.

La neuroplasticità è definita come l'abilità del sistema neurale di modificarsi strutturalmente e funzionalmente in risposta a eventi fisiologici,

patologici o stimoli ambientali, sia durante lo sviluppo che durante il resto della vita. Questa avviene a diverse granularità da cambiamenti a livello cellulare (e.g. modifiche durante l'apprendimento), a processi su aree maggiori che coinvolgono la ri-mappatura della corteccia (e.g. come conseguenza ad un infortunio).

Ricerche scientifiche hanno dimostrato che avvengono importanti cambiamenti nel sistema neurale e che questi cambiamenti possono modificare, sia anatomicamente che fisiologicamente, gli schemi di attivazione neuronale come conseguenza dell'esperienza [13, 12].

Il Modulo Intenzionale è stato creato con lo scopo di adattarsi a diverse tipologie di segnali sensoriali. Se si inviano dati ricevuti da un sensore video, il Modulo Intenzionale si specializzerà su quel tipo di dato, ma, se decidiamo di usare un altro tipo di dato sensoriale (e.g. il suono percepito) con lo stesso Modulo Intenzionale, questo perderà la precedente specializzazione sui segnali provenienti dall'apparato visivo e si adatterà ai nuovi dati.

Un'altra importante funzione del modulo consiste nella capacità di saper sviluppare autonomamente nuovi obiettivi e nuove motivazioni. Questa funzione del modulo intenzionale è stata già provata con una configurazione di robot molto semplice, mirando allo sviluppo di nuove motivazioni e controllando lo sguardo, rappresentato da una fotocamera, verso nuove classi inaspettate di stimoli visivi [42, 44, 43]. Successivamente è stato provato che il modulo può apprendere autonomamente nuovi stimoli, come passare dall'apprezzamento istintivo di un colore al giudicare interessante la forma che lo contiene [9].

Per soddisfare tutte le funzioni descritte, il Modulo Intenzionale presenta al suo interno due strutture atomiche: il Modulo Ontogenetico, che ha il compito di sviluppare nuovi obiettivi, e il Modulo di Categorizzazione, che si occupa di dividere in aree di interesse gli stimoli sensoriali.

Inizialmente entrambi i moduli interni, Categorizzazione e Ontogenetico, sono vuoti. I dati sensoriali vengono trasmessi al Modulo di Categorizzazione che svolge la funzione di creare le categorie di interesse. Una volta che queste sono state realizzate, i dati vengono passati al Modulo Ontogenetico. Questo modulo, attraverso l'apprendimento Hebbiano [29], genera nuovi interessi e restituisce un segnale (che chiameremo segnale ontogenetico) che indica quanto questi nuovi interessi siano soddisfatti; valori alti di questo segnale corrispondono a un'alta soddisfazione rispetto agli interessi sviluppati. Il Modulo Intenzionale riceve in ingresso anche un segnale dal Modulo Filogenetico e restituisce in uscita il massimo tra il segnale del Modulo Filogenetico e quello generato dal Modulo Ontogenetico. Questo terzo segnale viene chiamato Segnale Rilevante e viene usato in due modi: serve

al Modulo di Categorizzazione per sapere quando deve apprendere e viene anche propagato nei Moduli Intenzionali successivi, come mostreremo nella descrizione dell'architettura completa.

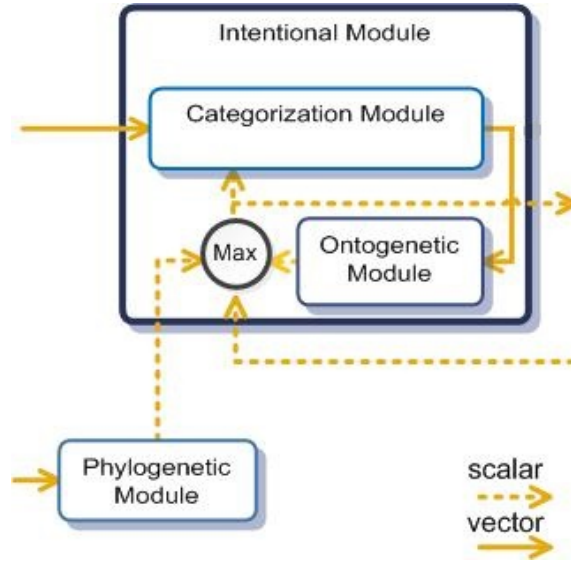


Figura 4.1: Modulo Intenzionale.

#### 4.3.1 Modulo di Categorizzazione

La categorizzazione è un'operazione cognitiva interessante nella quale il cervello raggruppa insieme oggetti che condividono tra di loro alcune proprietà, nonostante il loro aspetto sia differente. Ad esempio, l'essere umano istintivamente raggruppa insieme cani, gatti, pesci, uccelli e serpenti nella categoria "animale", anche se dal punto di vista visivo sono molto differenti [66]. Nel cervello le aree principalmente coinvolte nel processo di categorizzazione sono la corteccia pre-frontale e quella temporale inferiore. Queste due aree, quando interagiscono tra di loro, permettono di immagazzinare e richiamare le categorie [16]. Il modulo utilizzato per svolgere il processo appena presentato è il modulo di categorizzazione. Questo, quindi, rappresenta la corteccia cerebrale dell'architettura. Il modulo, dopo aver ricevuto un segnale dall'apparato sensoriale, inizia il processo di divisione in categorie. Per eseguire questa operazione, il segnale in ingresso viene elaborato due volte: inizialmente attraverso l'Analisi delle Componenti Indipendenti (Independent Component Analysis, ICA, vedere Appendice A) e successivamente attraverso un algoritmo di raggruppamento, K-Mean.

L'Analisi delle Componenti Indipendenti permette al modulo di generalizzare la rappresentazione del segnale in ingresso indipendentemente dal tipo di stimolo (e.g. audio, video, motorio, etc).

Il motivo per cui usiamo una rappresentazione di questo tipo è facilmente intuibile grazie ricerche che sono state effettuate sulla codifica delle informazioni reattive al nostro cervello. La codifica neuronale consiste nel tradurre uno stimolo in ingresso in un'attivazione neurale; quindi, una rete di neuroni potrebbe idealmente rappresentare ogni tipo di informazione in ingresso. Infatti, immaginando che le informazioni in ingresso possano essere di vario genere (e.g. sensoriali e non), il fatto che vengano tradotte a prescindere dalla tipologia, nonostante le differenze, sottoforma di attivazioni neurali, fa supporre che i neuroni siano in grado di codificare ogni tipo di informazione[65].

Il raggruppamento avviene per somiglianza: il segnale viene incluso in una determinata categoria quando può essere associato ad uno dei gruppi già presenti nel sistema. Nel momento in cui, invece, non si riesce ad attribuirgli una categoria, allora si procede con la creazione di un apposito gruppo contenente il nuovo segnale e tutti quelli simili che verranno poi analizzati.

Come detto in precedenza, la categorizzazione ha luogo unicamente se il sistema ritiene che ciò che sta percependo è interessante, cioè il Segnale Rilevante deve essere sopra la soglia predeterminata per l'attivazione.

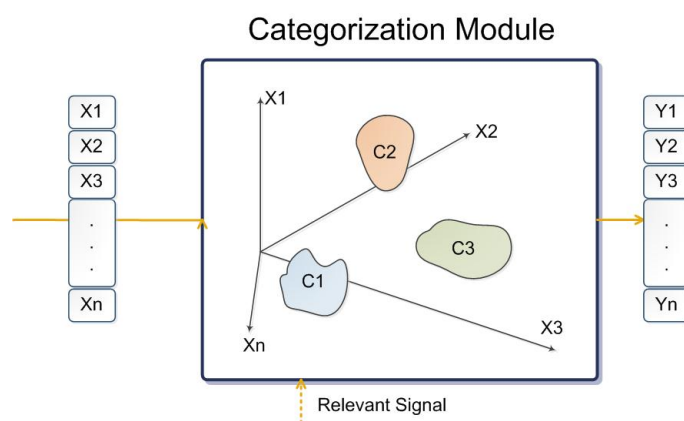


Figura 4.2: Modulo di Categorizzazione.

### 4.3.2 Modulo Ontogenetico

Il talamo è composto da un gruppo di cellule, o nuclei, posizionato centralmente nel cervello e svolge un ruolo fondamentale nella gestione del flusso di informazioni dirette alla corteccia cerebrale [60]. Ogni porzione del talamo

è collegata reciprocamente con le parti equivalenti della corteccia cerebrale, per mezzo delle quali la corteccia può modificare le funzioni talamiche. Queste connessioni sono utilizzate soprattutto per le comunicazioni dalla corteccia al talamo piuttosto che viceversa [47].

Si può dedurre che, tramite questi collegamenti tra le due aree del cervello e la loro interazione, la generazione degli obiettivi nasca nel talamo e si diffonda attraverso tutta la corteccia cerebrale [67].

Nella nostra architettura il modulo adibito allo svolgimento dei compiti appena descritti è il Modulo Ontogenetico. Questo ha, quindi, il compito di analizzare l'uscita del Modulo di Categorizzazione per mostrare quanto il segnale in ingresso soddisfi gli obiettivi sviluppati. Per far ciò utilizziamo una funzione di apprendimento Hebbiano [29], applicata ad una batteria di percettroni [55] della stessa dimensione del massimo numero di gruppi creati dal Modulo di Categorizzazione, la quale restituisce il valore di interesse dei singoli neuroni attivati dal processo. Il valore più alto viene selezionato e trasmesso come Segnale Ontogenetico.

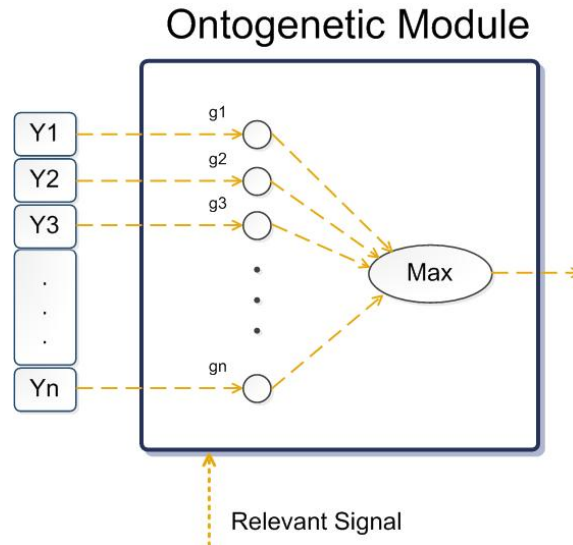


Figura 4.3: Modulo Ontogenetico.

## 4.4 ODIN

Il nostro cervello, ogni volta che riceve uno stimolo sensoriale, scompone questo segnale per farlo analizzare da diverse aree. Così facendo, non si analizza la totalità del segnale ma si studiano prima le componenti che lo costituiscono e poi, dopo averle studiate separatamente, si ricompongono per poterle studiare nel loro insieme (come in Figura 2.2).

Il cervello umano è in grado di risolvere problemi computazionalmente differenti grazie ad un grande numero di aree funzionali. Anche se questo numero elevato di aree specifiche lascia pensare che sia presente un meccanismo indipendente per ogni problema, si è visto che alla base di tutte queste aree si trovano dei meccanismi computazionali comuni. Questi meccanismi si sviluppano a partire da una parte computazionale comune che sottostà a ogni area funzionale: la rete neurale. Due di questi meccanismi sono la normalizzazione [11] e il population coding [56, 4]. Il primo consiste nel mediare la somma delle risposte delle attività della popolazione di neuroni, dividendo tale valore per il numero di neuroni coinvolti, allo scopo di codificare una rappresentazione neurale. Il population coding, invece, consiste nel codificare le informazioni sensoriali attraverso un gruppo di neuroni. Ogni neurone risponde a determinati stimoli e la risposta del gruppo osservato può essere analizzata per trovare nuove informazioni relative al segnale iniziale. Come descritto nel Capitolo 2 una interessante teoria, recentemente proposta da Kurzweil [39] e basata sul principio del population coding, sostiene che creando un gruppo di circa 100 neuroni sia possibile riconoscere qualsiasi schema.

Nella nostra architettura ogni modulo intenzionale si comporta come un piccolo agglomerato di neuroni. Quindi per replicare il population coding, abbiamo deciso di fare in modo che ogni strato di ODIN contenga al suo interno un numero di Moduli Intenzionali arbitrari come descritto dalla Formula :

$$|IM_x| = \sum_{j=1}^{|IM_{(x-1)}|} \binom{|IM_{(x-1)}|}{j} \quad (4.1)$$

dove  $|IM_x|$  rappresenta il numero di Moduli Intenzionali presenti nello strato  $x$  calcolati a partire dal livello precedente.

Per quanto riguarda, invece, la normalizzazione abbiamo deciso di considerare i segnali conseguenti l'attivazione dei Moduli Intenzionali dell'ultimo strato e mediarli tra di loro, così da ottenere un valore normalizzato prodotto a partire da un singolo segnale in ingresso.

Le connessioni dei moduli nel nostro sistema si basano su una rete feedforward parzialmente connessa a tre strati.

Il primo strato è costituito da due Moduli Intenzionali, ciascuno con un filtro: tale filtro può essere uguale o diverso a seconda delle situazioni e del compito richiesto. Il secondo blocco è formato da tre moduli, uno per ogni possibile configurazione degli ingressi, in modo da poter esaminare tutte le combinazioni, così come avviene nel cervello, e propagare in avanti tali informazioni. L'ultimo livello è invece composto da sette moduli, anche in questo caso connessi per soddisfare tutte le combinazioni possibili. Da questo ultimo livello estraiamo i risultati finali applicando il metodo di normalizzazione.

La rete potrebbe essere ampliata andando ad aggiungere altri livelli e connettendoli con quello precedente con lo stesso criterio visto in precedenza. Tuttavia le dimensioni del sistema aumenterebbero significativamente rendendo complessa la sua gestione e ingente la memoria richiesta. Infatti se consideriamo l'aggiunta di un quarto strato dovremmo aggiungere 127 nuovi moduli, come mostrato in (4.2), mentre l'aggiunta di un quinto livello ne introdurrebbe, come calcolato in (4.3), oltre un miliardo .

$$\binom{7}{1} + \binom{7}{2} + \binom{7}{3} + \binom{7}{4} + \binom{7}{5} + \binom{7}{6} + \binom{7}{7} = 7 + 21 + 35 + 35 + 21 + 7 + 1 = 127 \quad (4.2)$$

$$\binom{127}{1} + \binom{127}{2} + \binom{127}{3} + \dots + \binom{127}{126} + \binom{127}{127} \approx 1,7 * 10^{38} \quad (4.3)$$



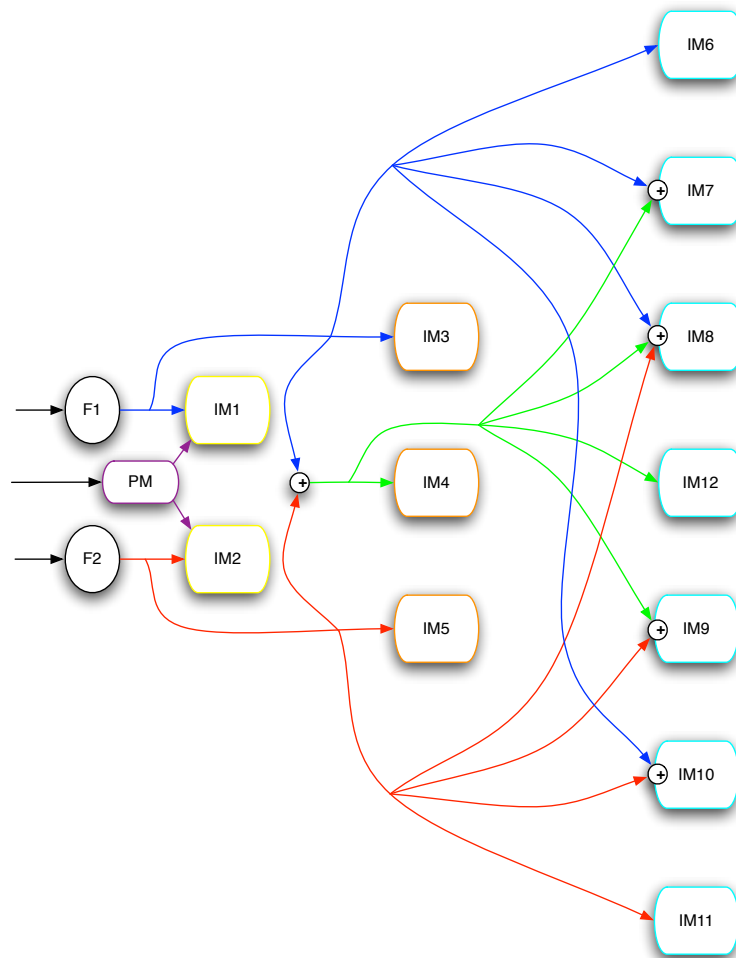


Figura 4.4: Architettura ODIN senza la propagazione dei segnali ontogenetici.

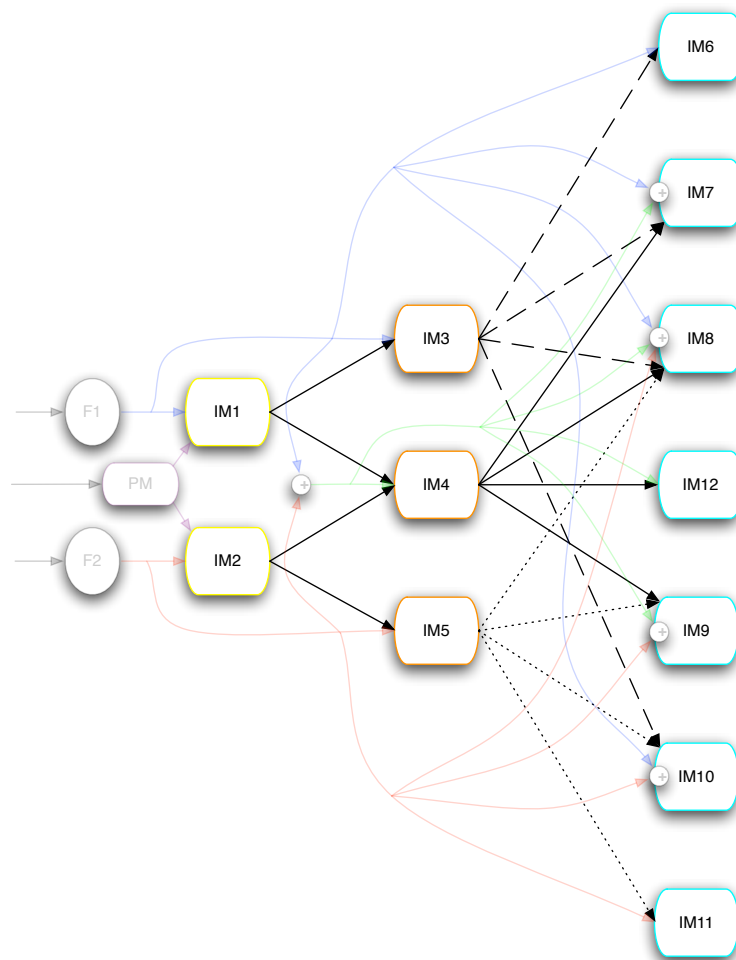


Figura 4.5: Architettura ODIN con la propagazione dei segnali ontogenetici.

## 4.5 Il modello del sistema motorio

Poiché uno degli sviluppi futuri più interessanti di questo progetto è l'implementazione bioispirata del movimento, andando ad unire questa funzionalità del cervello con quelle già presenti nel progetto, abbiamo deciso di creare un sistema capace di riprodurre il funzionamento della parte motoria del cervello modellizzando le parti interessate: cervelletto, gangli della base e corteccia.

Abbiamo, quindi, studiato e creato un modello che utilizza e fonde insieme la parte riguardante l'interesse (amigdala, talamo e corteccia) con la parte motoria. Per quanto possa sembrare strano, le due funzioni sono collegate tra loro, infatti, oltre alla presenza di parti del cervello comuni a entrambe, le due aree svolgono una funzione di supporto reciproco. Il movimento si genera in seguito alla necessità di trovare o raggiungere qualcosa di interessante.

La parte che studia gli interessi, amigdala e talamo, riceve le informazioni che vengono trasmesse dai vari organi di senso e, dopo averle analizzate, trasmette l'informazione alla corteccia cerebrale. La corteccia, oltre a ritrasmettere le informazioni alle altre aree del cervello che non intervengono nella selezione del moto e che quindi non analizzeremo, crea una rappresentazione sensoriale dell'ambiente, elabora l'azione da compiere e comunica la scelta alle altre due aree alle quali compete la gestione del movimento: gangli della base e cervelletto. I primi sono i responsabili del controllo della traiettoria durante i movimenti volontari e apportano correzioni durante l'esecuzione dell'azione stessa, mentre il cervelletto valuta l'effettivo raggiungimento dell'obiettivo e, in caso negativo, segnala la differenza tra posizione raggiunta e posizione attesa. Per comunicare i cambiamenti di direzione o le nuove posizioni da raggiungere, sia i gangli della base sia il cervelletto comunicano con il talamo che poi trasferisce queste informazioni alla corteccia. Poiché gangli e cervelletto non possono comunicare direttamente tra loro, la corteccia e il talamo svolgono anche il compito di traduzione e consegna dei rispettivi messaggi.

Da un punto di vista di apprendimento, le tre aree coinvolte si comportano seguendo diversi metodi: apprendimento non supervisionato per la corteccia, apprendimento supervisionato per il cervelletto e apprendimento per rinforzo per i gangli basali. Risulta, infatti, che la gestione del moto viene svolta da queste aree che si confrontano e correggono a vicenda: la corteccia crea un modello del mondo e seleziona le azioni possibili da svolgere, i gangli della base selezionano quella più consona in base all'ambiente circostante e ne valutano l'esecuzione tramite una funziona di ricompensa;

infine il cervelletto valuta la bontà dell'esecuzione del compito [22, 21]. In particolare sembra esserci evidenza che i gangli possano essere modellizzati con l'actor-critic, un particolare caso di apprendimento per rinforzo in cui è presente un esecutore, actor, che svolge un compito (e.g. raggiungere un oggetto all'interno di una mappa) e un supervisore, critic, che ha il compito di osservare le azioni dell'esecutore e dare una valutazione alle scelte compiute [31, 8]. La parte del supervisore viene svolta dal cervelletto, facendo così cooperare le due aree.

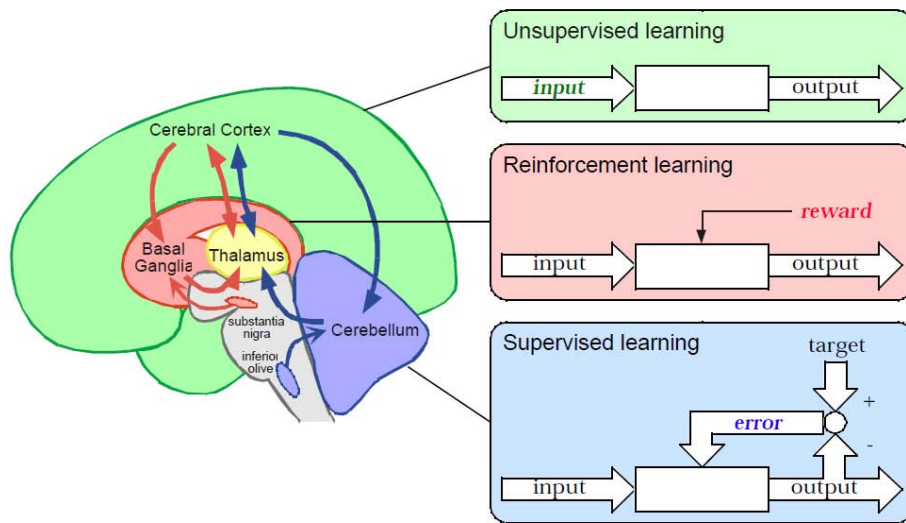


Figura 4.6: Aree di apprendimento nel cervello come presentato da Kenji Doya [21, 22].

L'azione, quindi, si svolge in questo modo: la corteccia sensoriale ricostruisce l'ambiente e la posizione del corpo e trasmette l'informazione ai gangli e al cervelletto. Se l'azione seguente è un'azione conosciuta, la parte mediale del cervelletto si attiva e comunica, tramite la corteccia, ai gangli della base che l'azione da compiere è già stata immagazzinata all'interno dell'area chiamata *matrice*. A questo punto vengono estratte le informazioni del movimento da eseguire e sono quindi passate dai gangli alla corteccia motoria, che esegue il movimento. Se invece l'azione non è conosciuta si attiva la parte laterale del cervelletto che stima quale sarà lo stato raggiunto dopo l'esecuzione del movimento; la previsione è comunicata alla corteccia premotoria che informa i gangli della base sia dello stato atteso sia del movimento da compiere per raggiungerlo. I gangli, quindi, eseguono le correzioni e la valutazione delle traiettorie migliori per il raggiungimento dello scopo e comunicano alla corteccia motoria il movimento da compiere. Quando l'azione motoria arriva termine, la corteccia analizza nuovamente l'ambiente

circostante e trasmette l'informazione al cervelletto che valuterà l'effettiva bontà del movimento.

Poiché la parte di movimento può essere validata sia attraverso un simulatore sia grazie ad un sistema robotico reale (e.g. NAO Robot), il modello che abbiamo sviluppato prevede l'implementazione delle primitive motorie. Come detto in precedenza, secondo studi recenti, le azioni e i movimenti sono composti da blocchi elementari chiamati primitive motorie. Queste primitive possono essere trasformate e combinate in modi differenti per ottenere l'intero repertorio di azioni complesse per un organismo. Possiamo considerare le primitive motorie come l'attivazione di un singolo muscolo nel tempo; più il valore sarà elevato, più il muscolo sarà attivato portando a una maggiore velocità nell'esecuzione finale del gesto. Per utilizzare le primitive, quindi, bisogna introdurre dei fattori di scala grazie ai quali si può comandare l'intensità dell'attivazione del muscolo. Il nostro modello di sistema motorio prevede il calcolo e il salvataggio dei pesi. All'inizio i pesi sono calcolati e aggiornati in ogni iterazione, così che possa imparare e apprendere le azioni corrette da replicare in situazioni similari. Ogni volta che il sistema porta a termine un'azione che il cervelletto reputa corretta, i pesi utilizzati per compiere il movimento vengono salvati nei gangli della base, più in particolare nella matrice. Ogni movimento eseguito è in realtà composto da micro-azioni che permettono di raggiungere i punti di via necessari per portare a termine l'azione completa. Grazie alla funzione di categorizzazione dei movimenti del sistema queste micro-azioni vengono salvate e organizzate in cluster. Il concetto alla base della scelta di produrre un'azione nota o di creare un nuovo movimento, è la vicinanza della proiezione della micro-azione appena svolta ad uno dei centroidi dei cluster: se la distanza tra il punto e il centroide è inferiore rispetto a una soglia fissata a priori, allora il sistema attiverà la parte dei movimenti noti, andando a leggere i pesi da assegnare alle primitive motorie direttamente dalla matrice.

Di questo modello motorio non sarà presentata nessuna implementazione. Come detto in precedenza, lo sviluppo di questo modello è stato propedeutico allo sviluppo dell'architettura ODIN.

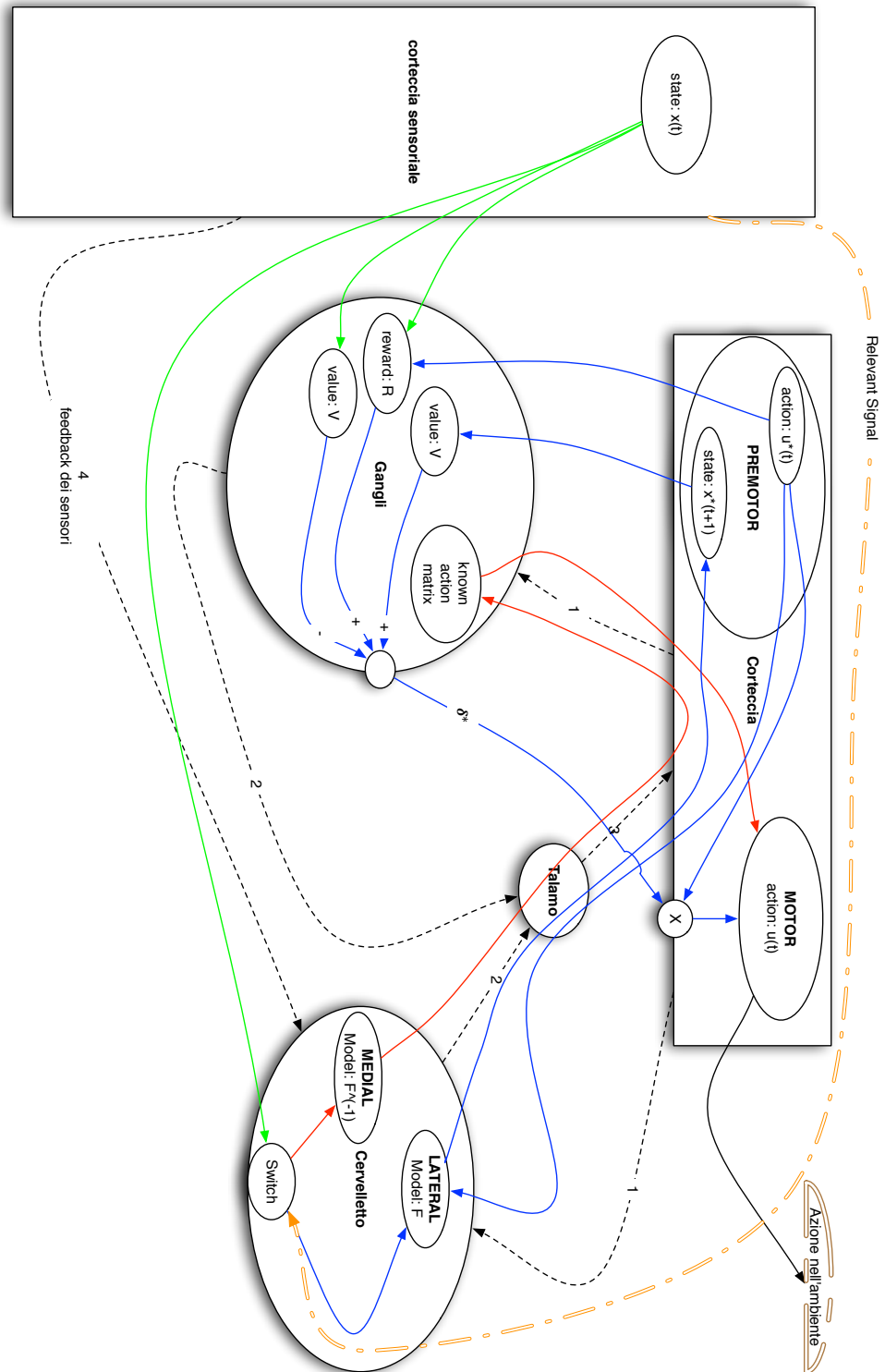


Figura 4.7: Modello del sistema motorio.

## Capitolo 5

# Implementazione

*“Luke: All right, I’ll give it a try.*

*Yoda: No. Try not. Do...or do not. There is no try”*

Luke and Yoda, Star Wars

In questo Capitolo presentiamo le scelte implementative e le metodologie utilizzate che hanno portato alla realizzazione del sistema descritto nel Capitolo 4. Alcune parti sono state implementate integralmente, mentre altre sono state ottenute reingegnerizzando ciò che era stato precedentemente sviluppato. In particolare, come strumento di modellizzazione e implementazione, abbiamo utilizzato MATLAB. Questo ha richiesto un lavoro di conversione del progetto di partenza, poiché è stato necessario trasformare il codice scritto in C# in codice leggibile dallo strumento MATLAB.

### 5.1 Filtraggio degli Ingressi

I filtri che abbiamo utilizzato si dividono in due categorie: audio e video. Ricevuta in ingresso un’immagine colorata, quello che viene fatto dal sistema è creare tre istanze della stessa figura, una contenente la saturazione dei singoli pixel, una in bianco e nero e una che contiene solo i bordi degli oggetti presenti. Per ottenere le prime due siamo ricorsi al metodo di trasformazione da immagine RGB (Red Green Blue) a quella HSV (*Hue Saturation Value*, vedere Figura 5.1) che permette di ottenere facilmente sia la saturazione, essendo il secondo valore fornito dalla trasformazione, sia il bianco e nero, poiché il Value rappresenta la tonalità di grigio del pixel esaminato.

Questi procedimenti sono stati implementati facilmente grazie ad una funzione già presente nella libreria standard di MATLAB, chiamata **rgb2hsv**. Per quanto riguarda, invece, la trasformazione di una immagine RGB in

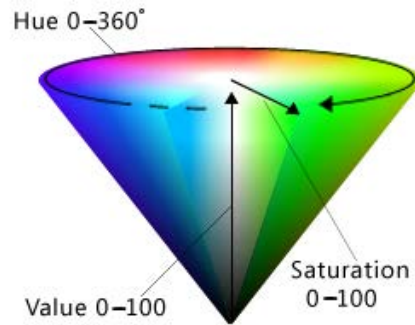


Figura 5.1: Visuale a cono della rappresentazione HSV.

un'immagine che mostra solo i contorni degli oggetti presenti, abbiamo utilizzato un filtro di Canny [10], che è uno dei metodi più utilizzati per eseguire il riconoscimento dei bordi. In MATLAB è presente una funzione che permette nativamente questa trasformazione: la funzione `edge` con l'aggiunta del parametro `canny` (vedere Figura 5.2).

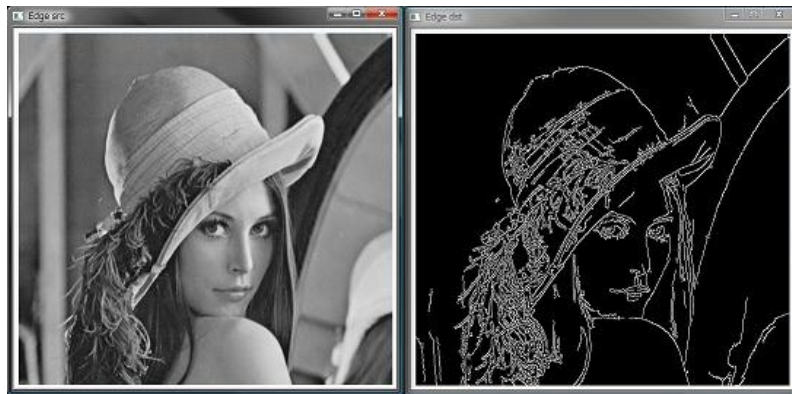


Figura 5.2: Esempio di immagine a cui è stato applicato il filtro di Canny.

In presenza di segnali audio abbiamo deciso, invece, di utilizzare la trasformata veloce di Fourier (*Fast Fourier Transformation*, FFT) prendendone la potenza e centrandola (vedere Figura 5.3). In questo caso sono stati necessari alcuni semplici passaggi con funzioni già disponibili nelle librerie standard.

A prescindere dal tipo di segnale in ingresso, che è una matrice di dati, il segnale restituito sarà un vettore che verrà analizzato dai moduli successivi.



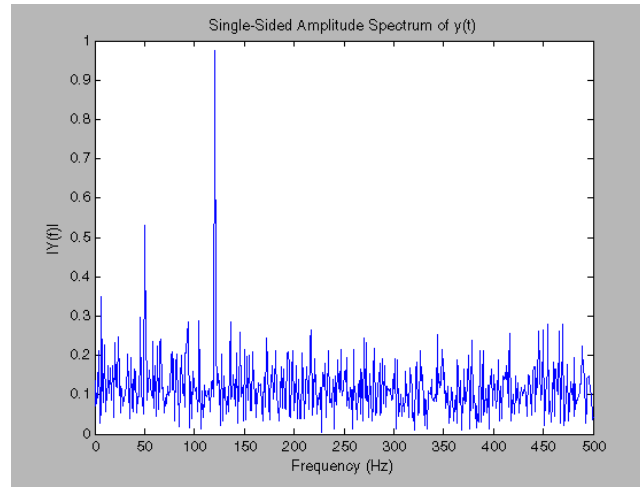


Figura 5.3: Esempio di segnale filtrato tramite FFT.

## 5.2 Modulo Filogenetico

Quando viene visualizzata una immagine, il Modulo Filogenetico genera un valore in uscita, chiamato Segnale Filogenetico, compreso tra zero e uno che rappresenta l'interesse per la figura osservata. Questo valore dipende direttamente dal quantitativo di colore presente all'interno dell'immagine osservata, dato che a questo modulo viene inviata l'immagine contenente solo la saturazione. Per calcolare questo segnale viene valutato il numero di pixel ritenuti interessanti, che abbiano cioè un valore di saturazione maggiore di una soglia impostata, in relazione al numero di pixel totali che compongono l'immagine stessa.

$$Phylosig = \frac{Pixel_{Colorati}}{Pixel_{Totali}} \quad (5.1)$$

Il valore così ottenuto viene utilizzato come segnale di controllo per la fase di apprendimento; quando viene superata la soglia impostata sarà attivata l'unità di apprendimento del Modulo Intenzionale, facendo in modo che il sistema impari solo quando c'è interesse nel farlo.

## 5.3 Modulo Intenzionale

Il Modulo Intenzionale è la parte fondamentale dell'architettura. Creato partendo dal modello ispirato al funzionamento della coppia talamo-corteccia, ha lo scopo di apprendere quando l'input che osserva è rilevante in base ai suoi comportamenti innati. Riceve in ingresso il segnale proveniente dai filtri e, dopo averlo analizzato, restituisce un numero compreso tra 0 e 1,

detto Segnale Rilevante. Questo valore stabilisce quanto l'informazione sensoriale sia importante, selezionando il segnale più rilevante tra quelli ricevuti dal Modulo Filogenetico, Ontogenetico e i Segnali Rilevanti di altri Moduli Intenzionali collegati dagli strati precedenti.

$$RelevantSignal = \max(Phylosig, Ontogenetic, otherRelevant) \quad (5.2)$$

Dove *RelevantSignal* è il Segnale Rilevante, *Phylosig* è il Segnale Filogenetico, *Ontogenetic* è il Segnale Ontogenetico e *otherRelevant* è la media dei Segnali Rilevanti degli altri Moduli Intenzionali collegati a quello preso in considerazione.

Al suo interno sono presenti altri moduli che svolgono delle funzioni specifiche: il Modulo di Categorizzazione e il Modulo Ontogenetico. Serve da contenitore e da supervisore affinché lo scambio dei dati avvenga correttamente.

### 5.3.1 Modulo di Categorizzazione

Il Modulo di Categorizzazione ha come scopo la divisione in gruppi degli ingressi del sistema. Quando viene ricevuto il segnale dai filtri sensoriali inizia la fase di proiezione degli ingressi. Il segnale in ingresso viene inizialmente proiettato come punto sullo spazio delle componenti indipendenti, ottenute precedentemente mediante fase di addestramento dell'algoritmo ICA, e viene elaborato il segnale di uscita del Modulo di Categorizzazione.

L'Analisi delle Componenti Indipendenti (ICA) permette al modulo di generalizzare la rappresentazione dell'ingresso, indipendentemente dal tipo di stimolo percepito. Nella fase precedente l'addestramento vengono estratte le componenti indipendenti da una serie di ingressi tramite l'algoritmo ICA. La proiezione dell'ingresso in questo spazio di componenti indipendenti serve per ridurre la dimensione dei dati e per avere una rappresentazione generalizzata del segnale:

$$\bar{W} = IC \times \bar{I} \quad (5.3)$$

Dove  $\bar{W}$  è il vettore dei pesi risultante (il punto proiettato),  $IC$  è la matrice delle componenti indipendenti e  $\bar{I}$  è il vettore in ingresso.

Dopo la proiezione dell'ingresso si passa all'analisi e alla composizione dell'uscita. Il vettore restituito da questo modulo veniva, in IDRA, anche utilizzato come uscita del Modulo Intenzionale; questo vettore poteva prendere come valori la distanza del punto proiettato dai centri dei gruppi, il codice binario del centro più vicino o le coordinate del centroide del gruppo a cui appartiene il punto proiettato. In origine, il vettore delle distanze del

punto proiettato dai centroidi dei gruppi presenti veniva trasferito, dopo essere stato restituito dal Modulo di Categorizzazione al Modulo Intenzionale, ai moduli successivi connessi come in Figura 4.5. Abbiamo avuto modo di valutare che questa soluzione, a prescindere dal contenuto del vettore trasmesso (sia con le distanze, sia con i codici binari che con i centroidi), non presentava abbastanza informazioni per permettere la creazione di nuovi gruppi basati su concetti più astratti.

La soluzione che abbiamo deciso di adottare, sia perché ritenuta valida da un punto di vista biologico sia perché consente di trasmettere una maggiore quantità di informazione agli strati successivi dell'architettura, è stata quella di inoltrare in ingresso a tutti i Moduli Intenzionali i vettori di immagine dei Moduli Intenzionali dello strato precedente ad essi collegati. In presenza di più ingressi, come mostrato in Figura 5.4, abbiamo deciso di sovrapporre idealmente le immagini facendo una media dei valori dei pixel corrispondenti.



Figura 5.4: Esempio di Moduli Intenzionali con più ingressi.

Questo metodo ci permette di mantenere le caratteristiche dei filtri iniziali andando ad esaltare le parti ritenute interessanti in entrambe le immagini e rendendo meno importanti le parti meno significative (e.g. dando così maggiore risalto ai pixel corrispondenti ai bordi dell'immagine in bianco e nero per mezzo della sovrapposizione con quella generata dal filtro di Canny).

Riprendendo l'analisi dell'uscita del Modulo di Categorizzazione, abbiamo deciso di mantenere come vettore in ingresso del Modulo Ontogenetico quello delle distanze dai centroidi del segnale in ingresso proiettato nello spazio delle componenti indipendenti. Questo vettore corrisponde, infatti, all'attivazione del neurone centrato in ogni gruppo:

$$y_i = \rho(x, C_i) \quad (5.4)$$

Dove  $y_i$  è la distanza dell'ingresso attuale dal centro del cluster  $i$ -esimo,  $x$  è l'input e  $C_i$  è il centroide dell' $i$ -esimo gruppo. Per quanto riguarda la nostra architettura i valori sono anche normalizzati tra zero e uno.

Successivamente sono valutati il Segnale Rilevante e la distanza del punto proiettato dai centroidi dei gruppi esistenti. Se il Segnale Rilevante risulta

essere sopra una certa soglia fissata a priori e se tutte le distanze dai centroidi sono inferiori ad un'altra soglia, sempre fissata a priori, il punto viene aggiunto al gruppo più vicino tramite l'algoritmo K-means. Se, invece, almeno una delle distanze dai centroidi risulta essere sopra la soglia, il punto viene aggiunto all'insieme dei punti interessanti e viene ricalcolata l'assegnazione alle categorie aggiungendo un nuovo gruppo, sempre tramite l'algoritmo di categorizzazione. Nel momento in cui il Segnale Rilevante risulta essere sotto soglia, il sistema reputa il segnale in ingresso non interessante e, quindi, non viene eseguita la procedura di categorizzazione.

### 5.3.2 Modulo Ontogenetico

Il Modulo Ontogenetico ha il compito di analizzare l'uscita del Modulo di Categorizzazione per sviluppare nuovi obiettivi e interessi. Per fare ciò il modulo si avvale anche di una funzione di apprendimento Hebbiano [29] implementata su una batteria di percettroni [55]; questa rete ha la stessa dimensione (in numero di percettroni) del massimo numero di gruppi creati dal Modulo di Categorizzazione.

Il valore degli attivatori neuronali in ingresso è valutato usando un vettore di pesi e il valore più alto viene selezionato e trasmesso come Segnale Ontogenetico:

$$O_s = \max_i(y_i w_i) \quad (5.5)$$

dove  $O_s$  è il risultante Segnale Ontogenetico,  $y_i$  è l'attivazione del neurone  $i$ -esimo e  $w_i$  è il vettore dei pesi associati al neurone  $i$ -esimo. Ogni peso viene poi normalizzato tra zero e uno. Nella vecchia architettura, una volta che un peso superava il valore fissato a priori di 0.8, veniva automaticamente modificato in modo da avere un valore unitario e rimaneva fissato per tutto il tempo di vita del sistema, a prescindere dal possibile valore reale che poteva essere raggiunto da quel parametro (e.g. abbiamo notato come alcuni pesi arrivassero sopra le centinaia). Dato che questa approssimazione non poteva più essere valida per un sistema più complesso come quello che abbiamo sviluppato, abbiamo deciso di modificare il metodo di normalizzazione. Ogni volta che sono calcolati i pesi, quelli non normalizzati sono salvati dal sistema; i pesi non normalizzati serviranno poi per il calcolo di quelli nuovi nell'iterazione successiva. La normalizzazione, quindi, avviene tramite la divisione di ogni peso per quello che risulta essere il maggiore tra tutti in quella iterazione. Avremo così un elemento sempre uguale ad uno e gli altri scalati in base alla loro importanza.

Il Segnale Ontogenetico, quindi, dipende molto dai pesi usati per valutare l'ingresso del modulo. Questi pesi, come detto in precedenza, vengono aggiornati durante ogni iterazione usando una funzione di apprendimento Hebbiano:

$$w_i = w_i + \eta(h_s y_i - (w_i y_i^2)) \quad (5.6)$$

Dove  $\eta$  è il tasso di apprendimento, mentre  $h_s$  è il segnale hebbiano che nel nostro caso non è altro che il Segnale Rilevante.

L'uscita di questo modulo è il Segnale Ontogenetico che rappresenta quanto lo stato dell'attuale ingresso soddisfi i nuovi interessi, sviluppati attraverso il processo di apprendimento hebbiano.

## 5.4 ODIN

ODIN è strutturato come una rete parzialmente connessa formata da strati di Moduli Intenzionali per un totale di dodici moduli, come mostrato nella Figura 4.5 del Capitolo precedente .

Per passare informazioni dagli strati iniziali a quelli successivi abbiamo pensato di utilizzare il Segnale Rilevante. Questo ci permette di comunicare ai moduli successivi quanto il segnale in ingresso appena analizzato presenti del contenuto ritenuto importante dagli strati precedenti. Il passaggio così ottenuto è rilevante per quanto riguarda la parte di addestramento dei moduli; infatti, poiché i moduli sono addestrati in ordine sequenziale di strato, i blocchi successivi usufruiranno della conoscenza acquisita dai propri genitori per apprendere in maniera più selettiva, focalizzandosi principalmente su ciò che è ritenuto più importante.

In fase implementativa questo è stato gestito tramite il passaggio della media dei Segnali Rilevanti dei moduli afferenti al modulo considerato. Come descritto nella sezione dedicata al Modulo Intenzionale, viene effettuata la scelta del massimo tra il segnale così ottenuto e il Segnale Ontogenetico per la creazione del Segnale Rilevante del modulo considerato.

Ottenuti i Segnali Rilevanti di ogni Modulo Intenzionale finale, effettuiamo una media su tutti i sette moduli, in modo da ottenere un valore unico. Il valore così originato rappresenta l'interesse, o grado di riconoscimento, che il sistema attribuisce al segnale sensoriale.

Abbiamo quindi creato dei *classificatori binari* per ogni oggetto che volevamo fosse riconosciuto dal sistema, si veda il Capitolo 6. Questo classificatore assegna un valore, sempre compreso tra zero e uno, ad ogni elemento in ingresso al sistema; questo valore viene usato per discriminare quali se-

gnali appartengono alla classe degli oggetti riconosciuti e quali non ne fanno parte.

Per calcolare l'efficienza dei classificatori binari bisogna avvalersi dell'introduzione di una soglia comune discriminatrice che permetta la suddivisione dei due gruppi. Questa soluzione si è dimostrata non ottimale a causa della disparità dei dati osservati (e.g. anche all'interno dello stesso tipo di segnale la complessità di quello che viene analizzato rende difficile la scelta di una soglia comune ottima, come nel caso di immagini di lettere e facce). Per ovviare a questo problema abbiamo deciso di valutare la bontà del sistema ODIN utilizzando un classificatore *One-Vs-All* (OVA) [54]. Questo prevede la creazione di tanti classificatori binari quante sono le classi che devono essere riconosciute ed effettua un confronto tra ogni classificatore per ogni risultato ottenuto. Poiché MATLAB non presenta ancora una buona gestione degli oggetti tramite le classi, invece di creare un sistema composto da più architetture ODIN (una per ogni oggetto da riconoscere), abbiamo deciso di mandare in esecuzione sequenzialmente un sistema alla volta. Questo è stato fatto tramite il salvataggio dei file di addestramento e il successivo e seriale caricamento nell'architettura. La selezione del gruppo di appartenenza avviene secondo la formula

$$f(x) = \arg \max_i f_i(x). \quad (5.7)$$

Si potrebbe pensare che la soglia di discriminazione debba essere abbastanza alta (e.g. sopra 0.5), tuttavia capita di avere riscontri bassi per segnali che il classificatore dovrebbe riconoscere. La classificazione ottenuta tramite il metodo OVA permette, come detto in precedenza, di ovviare a questo problema. Nonostante il classificatore corretto attribuisca un valore basso ad un segnale che appartiene alla sua categoria, confrontando il suo punteggio con quelli attribuiti da ogni altro classificatore, si nota come questo sia comunque il valore più alto ottenuto. Così facendo il sistema assegna un segnale di ingresso alla classe riconosciuta.

Nel Capitolo 6 forniremo poi alcuni esempi pratici di valori restituiti sia dal classificatore binario che dal classificatore OVA.

## Capitolo 6

# Risultati sperimentali

*“I’ve seen things you people wouldn’t believe...”*

Roy Batty, Blade Runner

Per verificare la validità della nostra architettura abbiamo svolto alcuni esperimenti di riconoscimento e classificazione; in particolare abbiamo condotto esperimenti su due differenti dataset di immagini e uno su dati audio. Lo scopo è dimostrare che il sistema è in grado di funzionare bene con diverse tipologie di segnali sensoriali (audio e video appunto), ma anche in presenza di segnali provenienti dallo stesso sensore di diversa complessità, come le immagini dei volti se paragonate a quelle più semplici delle lettere.

Vogliamo far notare che, in fase di addestramento, abbiamo deciso di impostare a priori il Segnale Filogenetico sul valore di 0.9. Poiché i dataset utilizzati sono in bianco e nero avremmo dovuto colorare tutte le figure che volevamo utilizzare per l’addestramento (e.g. per le lettere abbiamo utilizzato più di seicento immagini). Un Segnale Filogenetico alto, come visto in precedenza, serve al sistema per capire quando deve imparare poiché sta osservando qualcosa di interessante; in particolare, quando analizza delle immagini, il Segnale Filogenetico è alto quanto più un’immagine è colorata.

Questo tuttavia non ha inciso sui risultati in quanto, provando a colorare solo alcune immagini con colori particolarmente saturi, il Modulo Filogenetico restituiva un valore alto.

### 6.1 Motivazioni

I motivi per cui abbiamo deciso di svolgere gli esperimenti che proponiamo sono strettamente collegati alla tipologia di compito che vorremmo far

svolgere al nostro agente quando sarà implementata l'architettura motoria. L'analisi di un segnale video, discretizzato in immagini sequenziali, serve per il controllo e la ricerca degli obiettivi da raggiungere durante il movimento; risulta quindi fondamentale che tale elaborazione fornisca dei buoni risultati. La scelta del primo esperimento è ricaduta sul riconoscimento di lettere latine (OCR) per due motivi: da un punto di vista biologico è interessante, per una architettura bioispirata, il confronto con ciò che l'essere umano fa continuamente e con ottimi risultati, mentre, da un punto di vista ingegneristico, volevamo vedere se il sistema presenta delle performance accettabili in un campo ormai consolidato. Il secondo esperimento visivo si è concentrato sull'analisi dei volti perché, da un punto di vista biologico, il riconoscimento del viso è una delle prime attività riscontrabili fin dall'infanzia, mentre, da un punto di vista ingegneristico, risulta interessante testare il sistema proposto con immagini più complesse delle semplici lettere latine.

Oltre ai due esperimenti con le immagini, abbiamo deciso di effettuare un esperimento con un segnale audio, anch'esso utilizzabile successivamente nell'implementazione dell'architettura motoria, per valutare la possibilità di gestire diversi tipi di segnali sensoriali contemporaneamente, così come avviene nel cervello umano [38]. Per avere dei riscontri aggiuntivi sulla bontà del sistema, abbiamo deciso di utilizzare degli indici di classificazione statistica: *accuracy*, *precision* e *recall* [53].

L'*accuracy*, o accuratezza, rappresenta la percentuale di classificazioni corrette (positive e negative) rispetto al campione totale dei dati:

$$\frac{TP + TN}{TP + FP + TN + FN} \quad (6.1)$$

dove  $TP$  rappresenta i veri positivi (True Positive),  $TN$  indica i veri negativi (True Negative),  $FP$  si riferisce ai falsi positivi (False Positive) e  $FN$  denota i falsi negativi (False Negative).

La *precision*, o precisione, indica la percentuale di classificazioni positive e corrette ( $TP$ ) rispetto a tutte le classificazioni ritenute positive, comprendenti anche i falsi positivi:

$$\frac{TP}{TP + FP} \quad (6.2)$$

Il *recall*, o richiamo, rappresenta la percentuale di elementi correttamente riconosciuti del testing set rispetto a tutti gli elementi che sarebbero dovuti essere effettivamente classificati come veri positivi:

$$\frac{TP}{TP + FN} \quad (6.3)$$



La specificity, o specificità, indica la percentuale di elementi negativi reali correttamente classificati come tali:

$$\frac{TN}{TN + FP} \quad (6.4)$$

Abbiamo, infine, deciso di valutare le prestazioni del classificatore binario, utilizzato nel test con segnali audio, servendoci anche della curva ROC. Questa curva, che mette in relazione specificity e recall, permette di valutare la percentuale di successo della classificazione valutando l'area sottesa dalla funzione stessa (AUC).

## 6.2 Esperimento delle lettere

Come primo esperimento abbiamo scelto di verificare l'effettivo riconoscimento di lettere latine (Optical Character Recognition, OCR). Abbiamo quindi utilizzato un campione di 800 immagini per lettera, di dimensione  $128 \times 128$  pixel, che prevede solo caratteri maiuscoli ma con variazioni di font, applicazione del grassetto e del corsivo.



Figura 6.1: Esempio di diverse K

Abbiamo quindi suddiviso le immagini in due gruppi, addestramento e test, contenenti rispettivamente 640 e 160 elementi scelti casualmente. Le 640 immagini sono state utilizzate per addestrare il sistema mentre sulle restanti 160 abbiamo effettuato la prova di riconoscimento. Le immagini vengono prima analizzate tramite un filtro di individuazione dei bordi con il metodo di Canny e un filtro in bianco e nero; sia l'apprendimento che il riconoscimento si basano sui risultati del filtraggio appena illustrato.

Il test è stato svolto su dieci lettere scelte per alcune similitudini grafiche: A, B, D, E, I, K, P, R, S e T.

### 6.2.1 Risultati

Dopo la fase di addestramento abbiamo fornito al sistema le immagini da analizzare e per ognuna di esse abbiamo ottenuto un valore compreso tra 0 e 1. Questo valore rappresenta l'indice di gradimento della lettera osservata; l'immagine, infatti, viene paragonata mediante il processo di categorizzazione ad una delle classi che risultano essere interessanti (i gruppi ottenuti con K-Means) per il riconoscitore attualmente caricato sul sistema. Capita quindi

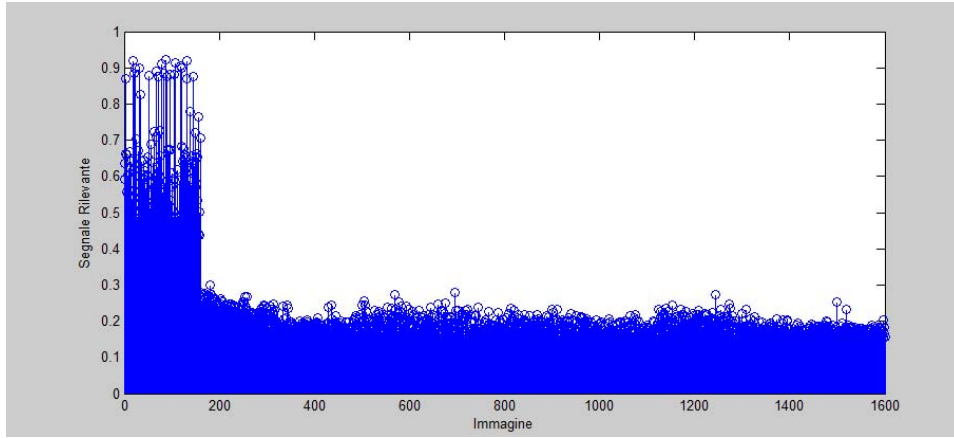


Figura 6.2: Risultati del Segnale Rilevante del classificatore della lettera A. Le prime 160 immagini corrispondono a quelle che il riconoscitore dovrebbe classificare correttamente mentre le successive 1440 sono di altre lettere.

di trovarsi con lettere a cui il sistema assegna un valore basso ( e.g. 0.3) quando, invece, dovrebbero avere un valore più alto perché sono effettivamente la lettera corretta.

Nonostante questo possa sembrare non corretto per un sistema di riconoscimento, in realtà non comporta alcuna perdita nelle prestazioni; infatti, all'inizio il sistema carica i dati per il riconoscimento della prima lettera (e.g. A) e analizza tutte le immagini restituendo un valore che rappresenta quanto il sistema sia confidente nel reputare che l'immagine osservata sia effettivamente quella attesa (e.g. una A). Subito dopo aver esaminato tutti i campioni vengono caricati i dati relativi all'addestramento di un'altra lettera (e.g. B) e ricomincia il processo di valutazione. Quando il sistema viene eseguito per tutte le lettere, si ottiene una matrice di valori compresi tra 0 e 1 le cui colonne rappresentano le immagini esaminate mentre le righe indicano i riconoscitori delle lettere, che sono organizzati in ordine alfabetico. Completata la matrice, si scorrono le colonne e si seleziona il valore più alto; la riga corrispondente rivela quale lettera è stata associata all'immagine visualizzata.

I risultati che abbiamo ottenuto facendo classificare 1600 immagini, 160 per ognuna delle 10 lettere riconosciute, mostrano 13 errori, con una percentuale di riconoscimento corretto del 99,18% (con un errore inferiore allo 0.8%), come si può vedere in Tabella 6.1. I riconoscimenti errati sono:

- 4 R erroneamente riconosciute come P

**R R P R**

- 2 P erroneamente riconosciute come R

**P P**

- 4 D erroneamente riconosciute come B

**D D D D**

- 1 B erroneamente riconosciuta come D

**B**

- 1 E erroneamente riconosciuta come B

**E**

- 1 T erroneamente riconosciuta come S

**T**

A partire dai dati presentati in Tabella 6.1, abbiamo calcolato l'accuracy, la precision e il recall del test.

Di seguito, nella Tabella 6.2, abbiamo riportato i valori dei tre parametri ottenuti per ogni riconoscitore di lettera. I risultati ottenuti risultano in media con quello che è il riconoscimento di caratteri stampati attualmente disponibile, andandosi ad attestare sopra il 97%[30, 64].

Lettera	(TP) True Positive	(FN) False Negative	(FP) False Positive	(TN) True Negative
A	160	0	0	1440
B	159	1	5	1435
D	156	4	1	1439
E	159	1	0	1440
I	160	0	0	1440
K	160	0	0	1440
P	158	2	4	1436
R	156	4	2	1438
S	160	0	1	1439
T	159	1	0	1440

Tabella 6.1: Risultati del test con le lettere.

Lettera	Accuracy	Precision	Recall	Specificity
A	100%	100%	100%	100%
B	99,63%	96,95%	99,38%	99,65%
D	99,69%	99,36%	97,5%	99,93%
E	99,94%	100%	99,38%	100%
I	100%	100%	100%	100%
K	100%	100%	100%	100%
P	99,63%	97,53%	98,75%	99,72%
R	99,63%	98,73%	97,5%	99,86%
S	99,94%	99,38%	100%	99,93%
T	99,94%	100%	99,38%	100%

Tabella 6.2: Precision, accuracy e recall del sistema con l'analisi delle lettere.

## 6.3 Esperimento volti

Dopo aver provato la funzionalità del sistema con immagini semplici, ci siamo concentrati su qualcosa di più complesso come il riconoscimento del viso. Ci siamo serviti di un dataset di foto di volti dell'università di Yale [1, 6] disponibile online per scopi di ricerca. In questo database sono presenti foto di diverse persone scattate in diverse condizioni di luce (Figura 6.3).



Figura 6.3: Esempio di foto dello stesso soggetto illuminate differentemente.

All'interno del dataset esistono due differenti gruppi, A e B: il primo è composto da foto che vanno dal collo ai capelli e che presentano anche elementi dello sfondo, mentre il secondo comprende solamente i volti, mostrando unicamente la parte dal mento alla fronte, come in Figura 6.4.

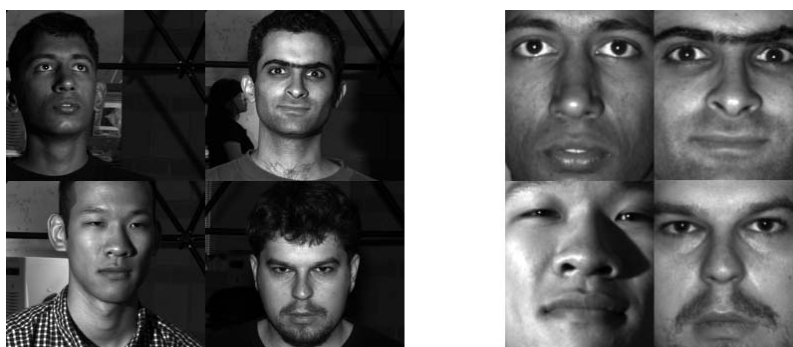


Figura 6.4: Esempi di immagini dal database A e le corrispettive del database B.

Noi ci siamo soffermati sul gruppo B, così da limitare sia la dimensione delle immagini (quelle dei soli volti sono  $168 \times 192$  pixel contro i  $640 \times 480$  di quelle complete) sia i disturbi dello scenario sullo sfondo che, oltre a non essere uniforme, varia da foto a foto. Abbiamo quindi svolto il solito ciclo di addestramento e test; tuttavia, visto il basso numero di immagini presenti per ogni soggetto, in fase di addestramento abbiamo utilizzato più volte le stesse immagini così da poter avere un sistema più robusto. Ovviamente le figure sulle quali è stata eseguita la fase di test non sono mai state usate per l'addestramento e, quindi, risultano del tutto nuove al sistema quando vengono analizzate.

### 6.3.1 Risultati

Utilizzando la classificazione OVA, abbiamo creato un classificatore per ogni persona. Ognuno degli 11 soggetti usati per l'analisi (Figura 6.5) ha 10 immagini di testing per un totale 110 immagini da analizzare.



Figura 6.5: Esempio di foto di ogni singolo soggetto utilizzato per l'esperimento.

Il sistema ha risposto particolarmente bene a questo test e, infatti, non abbiamo riscontrato errori nel riconoscimento. Questo risultato dipende fortemente da due aspetti legati al database impiegato: la complessità delle immagini e l'esiguo numero di campioni. Il primo si basa sulla diversità delle figure che vengono analizzate; infatti, diversamente dal test eseguito sulle lettere, il classificatore può ottenere informazioni più particolareggiate rispetto al semplice bordo e alla forma (e.g. la presenza di barba e altri elementi distintivi). Il secondo motivo è strettamente legato al dataset utilizzato; infatti, avendo a disposizione poche immagini, sia per il training che per il testing, è possibile che il sistema abbia memorizzato, invece di imparare, alcune peculiarità dei campioni.

Bisogna, tuttavia, considerare che il numero di immagini di testing è comunque esiguo e ciò comporta anche una minore possibilità di errore. Se avessimo avuto a disposizione un database con un numero di volti simile a quello delle immagini, avremmo potuto ottenere dei risultati più significativi sia per qualità che per quantità. Vogliamo comunque far notare che diversamente da altri lavori [57] non abbiamo modificato le immagini schiarendole o cercando di renderle più visibili, ma abbiamo mantenuto quelle originariamente fornite dall'università di Yale, scartando unicamente quelle controluce in cui si perdono i dettagli del volto (Figura 6.6).



Figura 6.6: Esempio di immagini scartate.

## 6.4 Esperimento audio

L'ultimo esperimento che abbiamo deciso di svolgere riguarda il riconoscimento di un suono. Lo scopo è di verificare che il sistema proposto sia effettivamente in grado di funzionare a prescindere dal tipo di segnale in ingresso così da poterlo poi applicare al modello motorio che abbiamo realizzato.

Non avendo a disposizione una raccolta di file audio abbiamo utilizzato delle tracce in varie lingue disponibili online [49]. In particolare, abbiamo utilizzato le registrazioni del vocabolo inglese “*sure*” come parola da riconoscere. Ogni registrazione della parola è stata ottenuta da persone diverse e con una differente intonazione (e.g. interrogativa, affermativa, crescente etc), proprio per come è stato creato il database originale. Come di consueto, abbiamo diviso il dataset nei due gruppi di training e testing, utilizzando rispettivamente 150 e 50 campioni. Per quanto riguarda le altre registrazioni utilizzate per il test, abbiamo usato altre tracce audio, sempre presenti sullo stesso sito, ma in lingue differenti.

Prima di mostrare i risultati, però, dobbiamo necessariamente fare una piccola premessa sui filtri usati e mostrare una limitazione architetturale presente: poiché abbiamo deciso di scartare la concatenazione, a causa degli scarsi risultati che produceva, le matrici che vengono analizzate, dopo l'applicazione dei filtri, devono necessariamente essere delle stesse dimensioni. Per quanto riguarda la gestione delle immagini il problema non si presenta; infatti, dopo l'applicazione dei filtri, l'immagine restituita rimane delle stesse dimensione di quella iniziale.

Il segnale audio, invece, necessita di essere trasformato prima di essere analizzato; queste trasformazioni modificano la dimensionalità delle matrici. All'inizio avevamo pensato di utilizzare l'ampiezza e la fase del segnale come filtri in ingresso, visto che queste due trasformazioni restituivano matrici con le stesse dimensioni; tuttavia, abbiamo visto che la fase del segnale delle varie tracce audio comportava dei peggioramenti delle prestazioni sistema, falsando così i risultati del test.

Abbiamo quindi ritenuto opportuno svolgere il test con uno stesso filtro applicato a entrambi i moduli iniziali e valutare se i risultati permettono di ritenere il sistema valido anche con un unico filtro applicato. Tutti i risultati ottenuti si basano quindi sul solo filtro di ampiezza applicato dopo una trasformata veloce di Fourier.

### 6.4.1 Risultati

Rispetto ai due esperimenti precedenti, essendo questo mirato alla verifica della sola possibilità di analizzare segnali diversi da quelli visivi, non abbiamo creato una batteria di classificatori ma ci siamo limitati ad un singolo riconoscitore binario, i cui risultati sono mostrati in Figura 6.7.

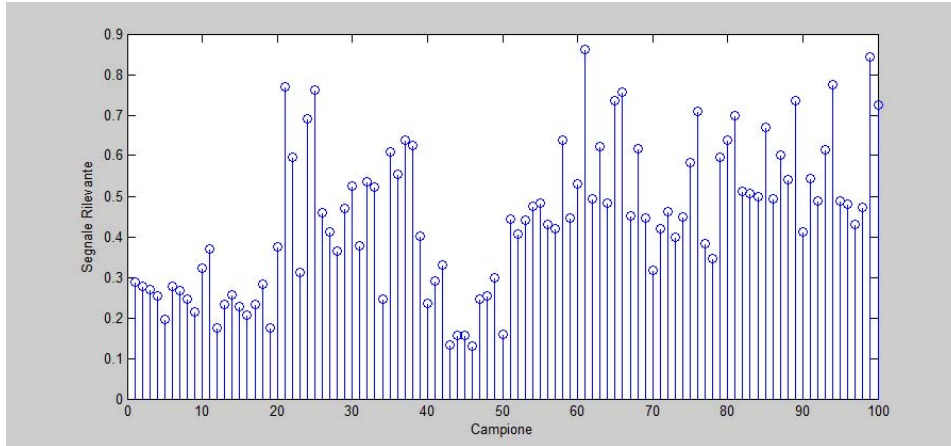


Figura 6.7: Risultati del test audio; i campioni che dovrebbero essere riconosciuti vanno dal 51 al 100.

Differentemente dagli altri esperimenti, non avendo utilizzato un classificatore OVA, abbiamo dovuto introdurre una soglia di discriminazione. Come detto in precedenza, trovare una buona soglia non è semplice; riportiamo, quindi, in Tabella 6.3 i risultati ottenuti con 5 diversi valori, in modo da considerare come soglia maggiore la più bassa che consente di non avere falsi positivi e come soglia inferiore la più alta che contiene tutti i veri positivi, con un passo discreto di 0.1.

Soglia	(TP) True Positive	(FN) False Negative	(FP) False Positive	(TN) True Negative
0.8	2	48	0	50
0.7	8	42	2	48
0.6	16	34	6	44
0.5	23	27	11	39
0.4	46	4	15	35
0.3	50	0	22	28

Tabella 6.3: Risultati del test audio.

Dalla Tabella 6.3 si può osservare come abbassando la soglia aumenti



il numero di falsi positivi; tuttavia, questo incremento risulta essere quantitativamente inferiore rispetto all'aumento dei veri positivi. Come nell'esperimento con le lettere, abbiamo voluto calcolare i parametri di accuracy, precision e recall relativi all'esperimento, visibili in Tabella 6.4.

Soglia	Accuracy	Precision	Recall	Specificity
0.8	52%	100%	4%	100%
0.7	56%	80%	16%	96%
0.6	60%	72.73%	32%	88%
0.5	62%	67.65%	46%	78%
0.4	81%	75.41%	92%	70%
0.3	78%	69.44%	100%	56%

Tabella 6.4: Accuracy, precision e recall del test audio.

Come si può osservare il valore che sembra corrispondere alla soglia migliore risulta essere 0.4. Questo non implica che 0.4 sia il valore più corretto in assoluto ma, nel nostro test con passo 0.1, risulta essere quello con i risultati migliori.

Abbiamo, quindi, calcolato la curva ROC dell'esperimento, visibile in Figura 6.8, per controllare che i risultati fossero soddisfacenti; l'area sottesa (AUC) è risultata essere circa il 78%, confermando che il test si può ritenere moderatamente accurato [25].

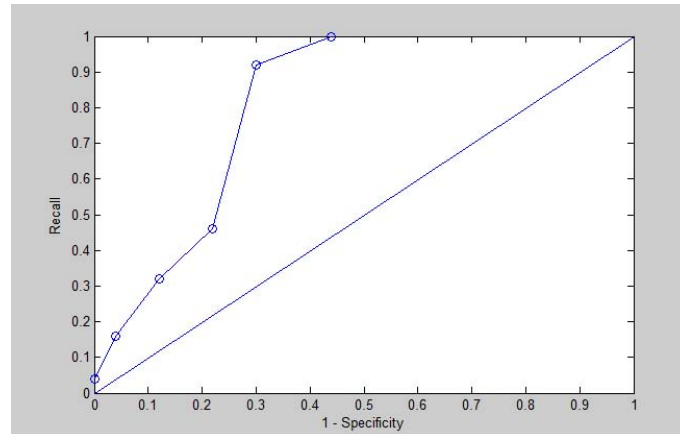


Figura 6.8: Curva ROC dei campioni audio.

Dobbiamo effettuare alcune considerazioni riguardante il dataset utilizzato, così come abbiamo fatto per i volti: una riguardante la durata dei campioni, l'altra relativa a come siano stati ottenuti questi ultimi. Poiché l'architettura necessita di matrici che abbiano la stessa dimensione, abbiamo

dovuto modificare le tracce audio disponibili poiché di diverse dimensioni e durata. Questo ha portato alla creazione di file leggermente più corti di altri che abbiamo dovuto modificare per portarli alle dimensioni prefissate tramite l'inserimento di silenzio. La procedura ha, quindi, creato dei campioni differenti rispetto alla maggioranza di quelli presenti; questo potrebbe aver influito sia in fase di training che di testing. La seconda considerazione riguarda il dataset stesso: i campioni sono stati registrati da persone diverse con timbri e intonazioni differenti.

## Capitolo 7

# Conclusioni e sviluppi futuri

*“To infinity and beyond!”*

Buzz Lightyear, Toy Story

Grazie a questa tesi abbiamo voluto creare un sistema bioispirato che fosse in grado di gestire e interpretare i segnali provenienti dal proprio apparato sensoriale. I robot basati su questo tipo di sistema, a differenza di quelli behaviour-based, non prevedono un insieme di obiettivi da raggiungere precedentemente integrati nella loro programmazione, ma consentono al robot stesso di svilupparne di nuovi mentre osserva il mondo circostante. L'unico aspetto determinato a priori sono i comportamenti innati, o istinti, che sono presenti anche in natura in ogni essere umano.

Recentemente è stata proposta un'architettura bioispirata che emula la parte cognitiva dell'essere umano, chiamata IDRA. Questa architettura mostra un'ottima gestione dello sviluppo di nuovi obiettivi a partire da semplici interessi preimpostati (e.g. l'apprezzamento di oggetti colorati), ma presenta qualche lacuna per quanto riguarda il sistema motorio. L'esperimento proposto, infatti, prevede solamente un semplice movimento dell'agente secondo una funzione pseudocasuale e non uno studio approfondito dei meccanismi biologici che sono alla base del movimento umano. Abbiamo quindi deciso di sfruttare l'idea di base di questa architettura relativa alla parte cognitiva per poter sviluppare un sistema percettivo bioispirato in grado di elaborare i segnali provenienti dall'apparato sensoriale che possa successivamente essere collegato con un modulo bioispirato per la gestione del movimento. Per valutare le caratteristiche che il nostro sistema deve implementare, abbiamo creato un modello che considera le funzioni e le interazioni tra le aree del cervello che controllano il movimento umano: il cervelletto, i gangli della base e la corteccia motoria. La creazione di questo modello complesso si

è resa necessaria poiché, per quanto le singole aree siano state studiate e documentate, nessuno ha proposto un sistema che le unificasse [21].

L'analisi di questo modello ci ha permesso di capire quali fossero le caratteristiche da considerare nella gestione della percezione e cosa avremmo dovuto rivalutare dell'architettura cognitiva IDRA. Questa reingegnerizzazione di quanto già presente e l'aggiunta della gestione dei segnali dagli input sensoriali, ci ha portati alla creazione del nostro sistema: ODIN. Tale sistema presenta una scelta bioispirata non solo per quanto riguarda la parte architettonica (e.g. la presenza di filtri per la scomposizione e l'analisi dei segnali in ingresso o il collegamento dei moduli funzionali secondo il *population coding*), ma anche per quanto riguarda gli algoritmi utilizzati per il riconoscimento. Abbiamo deciso, infatti, di implementare un classificatore OVA per emulare il comportamento dei neuroni, noto come *Winner-Take-All*, in cui il cervello risponde ad un determinato stimolo in base alle direttive del neurone che presenta il livello di attivazione maggiore.

Al fine di valutare le prestazioni e la bontà del nostro sistema, abbiamo condotto tre esperimenti: due riguardanti il riconoscimento di lettere e volti umani basati su un segnale proveniente dall'apparato visivo, e uno sul segnale proveniente dall'apparato uditivo, al fine di verificare se ODIN sia in grado di gestire segnali di differenti apparati sensoriali.

L'esperimento sul riconoscimento di caratteri (OCR) ci ha permesso di valutare il sistema ponendolo davanti a un compito svolto facilmente da un essere umano e che, anche in ambito ingegneristico, permette dei confronti con sistemi già presenti in letteratura al fine di controllare le prestazioni dell'architettura. I risultati ottenuti, 13 errori su 1600 lettere analizzate con una percentuale di riconoscimento corretto del 99.18%, si attestano sopra la soglia del 96 – 97%, ritenuta il valore che discrimina tra un riconoscitore appena sufficiente e uno buono [30, 64].

Il secondo esperimento che abbiamo proposto ci permette di verificare se il sistema è in grado di elaborare immagini più complesse rispetto alle singole lettere; abbiamo, infatti, voluto verificare se fosse in grado di riconoscere i volti di 11 persone attraverso i quali era stato precedentemente addestrato. I risultati ottenuti sono un ottimo indicatore della bontà del sistema ma non vanno presi come valore assoluto.

A causa del dataset utilizzato, composto da poche immagini, è possibile che il sistema abbia memorizzato i volti invece di riconoscerli, anche se i campioni di addestramento e di test sono sempre stati separati. Ulteriori prove con dataset più vasti permetteranno di calcolare le performance effettive del sistema, ma, da un punto di vista qualitativo, l'esperimento ci permette di sostenere che ODIN è in grado di gestire immagini complesse.

Infine, l'ultimo esperimento che abbiamo proposto riguarda l'analisi e il riconoscimento di un segnale uditivo. Abbiamo svolto tale esperimento per verificare la capacità del sistema di elaborare segnali provenienti da sensori differenti che supportino l'agente nei suoi movimenti e che non siano quelli visivi. I risultati di questo test ci permettono di affermare che ODIN riesce ad elaborare anche segnali differenti da quelli video, seppur con la limitazione di un solo filtro applicato al segnale entrante. Le prestazioni, misurate tramite lo schema grafico ROC, si attestano intorno al 78% di individuazioni corrette. Un risultato molto incoraggiante per gli sviluppi futuri che potranno essere presentati relativamente a questo sistema.

Tutti questi esperimenti ci permettono, quindi, di ritenere che ODIN sia un sistema valido per la gestione sensoriale secondo i parametri che ci eravamo posti all'inizio del lavoro di tesi. In futuro speriamo che l'architettura possa essere sviluppata maggiormente per portare avanti gli studi sulla creazione di un agente autonomo bioispirato. Di seguito mostriamo alcuni possibili sviluppi futuri che abbiamo individuato durante il nostro lavoro.

## 7.1 Sviluppi Futuri

Nonostante in questa tesi si siano ottenuti dei buoni risultati, questi sono ancora uno stadio preliminare di sviluppo rispetto alla realizzazione di agenti intelligenti capaci di adattarsi e confrontarsi con i cambiamenti dell'ambiente circostante. Possiamo identificare, infatti, alcuni sviluppi futuri atti al raggiungimento di tale obiettivo finale.

### Miglioramenti nell'elaborazione dei segnali

Abbiamo visto nel Capitolo 5 che i filtri alla base dell'architettura ODIN devono necessariamente restituire un vettore di informazioni di dimensione identica tra loro. Alla base di questa scelta c'è la possibilità di poter combinare i segnali in ingresso ai moduli dell'architettura in maniera bioispirata (e.g. sovrapponendo le immagini in bianco e nero con quelle filtrate tramite l'algoritmo di Canny), secondo quanto il cervello umano sembra effettuare biologicamente.

Una modifica interessante sarebbe quella di garantire la possibilità di avere un metodo di combinazione in grado di analizzare vettori con dimensioni differenti. Questo metodo, naturalmente, deve poter avere un collegamento con la sfera neuroscientifica dato che tutta l'architettura vuole essere una rappresentazione modellistica del funzionamento del cervello umano. La capacità di unire due segnali distinti, presi dalla stessa fonte, per riuscire ad

avere una rappresentazione del segnale reale, garantirebbe al sistema la scalabilità necessaria per la scelta dei migliori filtri da utilizzare.

### **Provare l'architettura con un sistema robot**

Come detto più volte in precedenza, lo scopo della tesi è quello di validare e implementare un modello capace di analizzare i segnali in ingresso in modo da poter gestire tutte le informazioni utili necessarie per una possibile integrazione con una architettura motoria. Quindi, in questo elaborato non abbiamo affrontato le problematiche di un test con l'ausilio di un robot. Le non idealità introdotte in questa tipologia di esperimenti (e.g. i colori non nitidi e la bassa risoluzione della telecamera, i disturbi nei segnali audio provenienti da altre fonti che andrebbero ignorate, etc) andrebbero studiate e analizzate. Il primo passo da affrontare sarebbe quello di convertire il codice scritto in MATLAB in un linguaggio di programmazione capace di interfacciarsi facilmente con un agente autonomo (e.g. *C#* per quanto riguarda il NAO Robot). Si potrebbe successivamente procedere cercando di ricreare un ambiente quasi ideale in cui provare le capacità del sistema per introdurre successivamente, di esperimento in esperimento, qualche elemento di disturbo e analizzare i cambiamenti di performance dell'architettura.

### **Implementazione del modello di movimento**

Come illustrato nel Capitolo 4, abbiamo creato un modello motorio tramite l'analisi neuroscientifica del funzionamento del cervello durante la pianificazione e l'esecuzione del movimento e la conseguente analisi della posizione in cui l'agente si trova. Questo modello è servito allo scopo di individuare le informazioni necessarie alla costruzione di una architettura percettiva in grado di supportarlo nel miglior modo possibile.

Un interessante sviluppo per questo lavoro di tesi sarebbe quello di implementare il modello sopracitato per provare effettivamente i collegamenti tra la parte sensitiva e quella motoria. Una volta realizzati dei test a supporto di un modello univoco (motorio e sensoriale appunto), sarebbe di grande utilità provare ad implementare tutta l'architettura all'interno di un robot antropomorfo autonomo, in modo da poter validare definitivamente il modello.

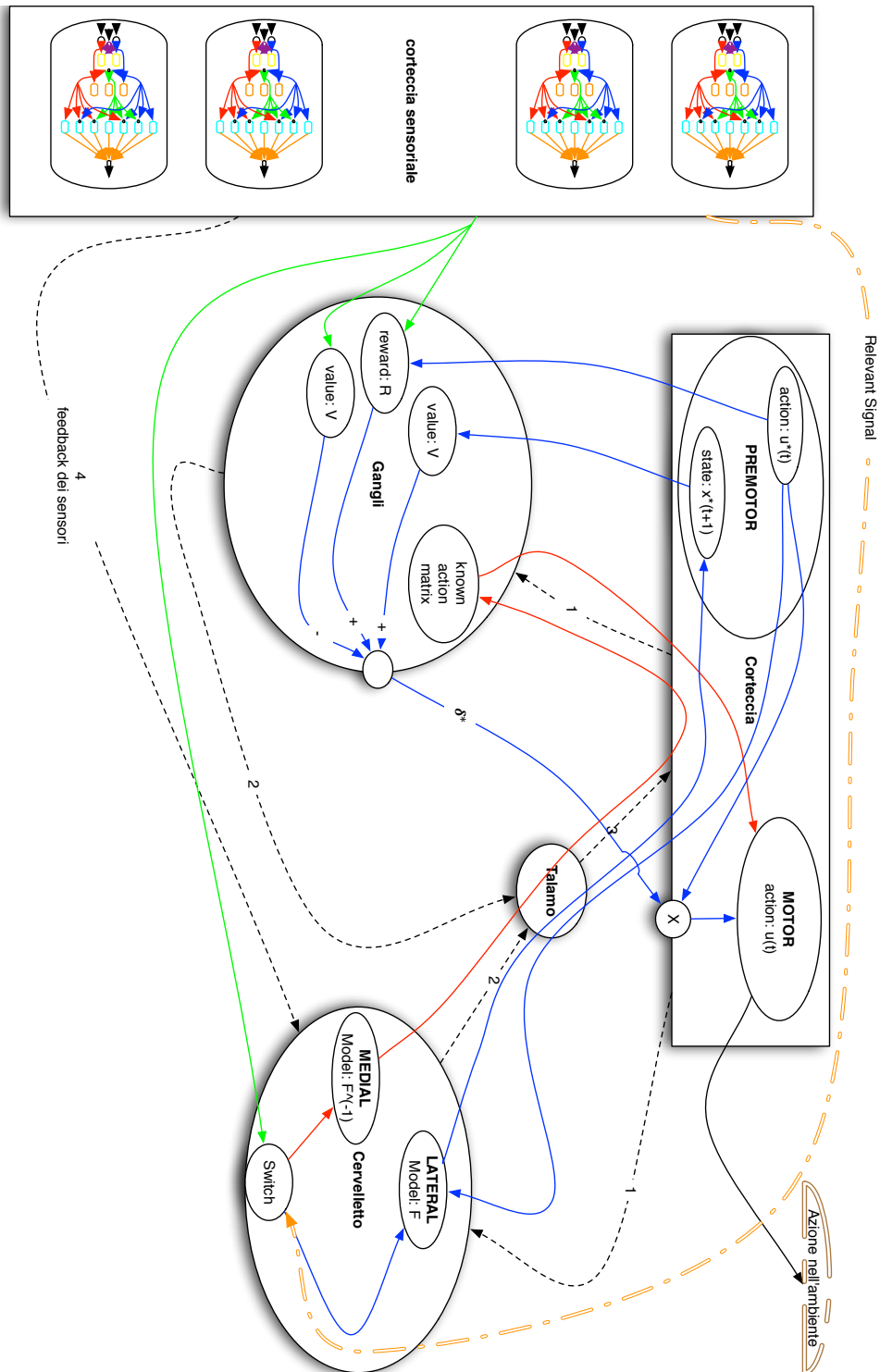


Figura 7.1: Un esempio di possibile modellizzazione della parte motoria associata a più sistemi ODIN.

### Ulteriore verifica delle prestazioni

Per validare il modello da un punto di vista sperimentale, in questa tesi sono stati proposti tre differenti esperimenti riguardanti due sfere sensoriali: i primi due relativi a quella visiva, mentre il terzo quella uditiva. Nonostante i risultati ottenuti siano buoni, potrebbero essere effettuate ulteriori investigazioni riguardanti le prestazioni del sistema. I dataset utilizzati, infatti, sono stati trovati disponibili gratuitamente in internet e, a esclusione di quello per le lettere, non si sono rivelati molto completi.

In primo luogo si potrebbe costruire un test per il riconoscimento delle parole, addestrando un sistema ODIN per ogni lettera e sottoponendo a tutti i moduli in sequenza le lettere componenti le parole stesse, come mostrato in Figura 7.2.

Un secondo esperimento molto interessante sarebbe quello di trovare o creare un database di volti e suoni più ampio di quello utilizzato negli esperimenti proposti. La possibilità di fare testing su un database ampio e completo porterebbe certamente ad avere dei risultati più precisi riguardo la capacità dell'architettura di riconoscere segnali differenti e più complessi rispetto a quello prodotto dalla visione delle lettere.

Per testare la capacità del sistema di gestire differenti tipologie di segnale, si potrebbero inoltre trovare o creare anche database di gesti motori. Questo esperimento risulterebbe particolarmente interessante per la combinazione del sistema ODIN con il modello motorio descritto nel Capitolo 4.



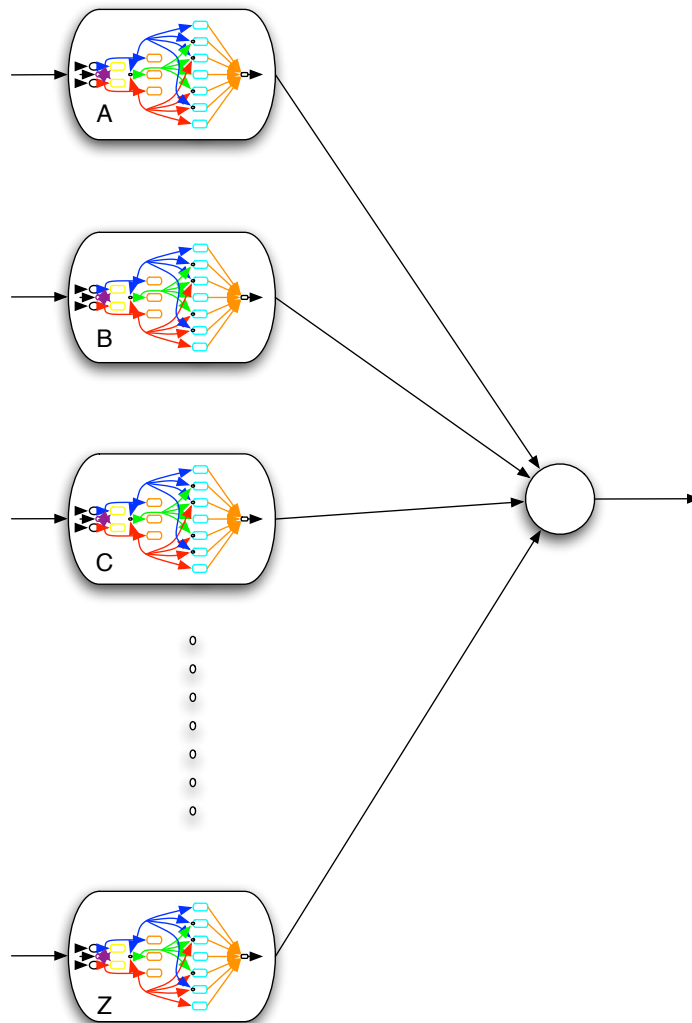


Figura 7.2: Possibile modello che implementi il riconoscimento delle parole. I vari moduli ODIN riconoscono la singola lettera (come specificato nella figura) e un sistema di classificazione successivo, che potrebbe anch'esso essere costituito da un nuovo sistema ODIN, impara le parole riconosciute.



# Bibliografia

- [1] Yale face database, 2007.
- [2] P. Abbeel A. Coates and A.Y. Ng. Apprenticeship learning for helicopter control. *Communications of the ACM*, 52(7):97–105, 2009.
- [3] N. M. Grzywacz A. L. Yuille. A winner-take- all mechanism based on presynaptic inhibition feedback. *Neural Computation*, 1:334–347, 1989.
- [4] A. B. Schwartz A. P. Georgopoulos and R. E. Kettner. Neuronal population coding of movement direction. *Science*, 233, 1986.
- [5] R. C. Arkin. *Behavior-based Robotics*. 1999.
- [6] P.N. Belhumeur A.S. Georghiades and D.J. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 23(6):643–660, 2001.
- [7] P. E. Latham B. B. Averbeck and A. Pouget. Neural correlations, population coding and computation. *Nature*, 7, May 2006.
- [8] A. G. Barto. Adaptive critics and the basal ganglia. pages 215–232. MIT Press, 1995.
- [9] M. Burrafato and L. Florio. A cognitive architecture based on an amygdala-thalamo-cortical model for developing new goals and behaviors: application in humanoid robotics. Master’s thesis, Politecnico di Milano, 2011-2012.
- [10] J. Canny. A computational approach to edge detection. *IEEE transactions on pattern analysis and machine intelligence*, PAMI-8(6), 1986.
- [11] M. Carandini and D. J. Heeger. Normalization as a canonical neural computation. *Nature Neuroscience*, 13, 2012.

- 
- [12] W. Chaney. *Workbook for a Dynamic Mind*. Houghton-Brace Publishing, 2006.
- [13] W. Chaney. *Dynamic Mind*. Houghton-Brace Publishing, 2007.
- [14] E. Booth Church. Why colors and shapes matter.
- [15] T. M. Cover and J. A. Thomas. Elements of information theory. 1991.
- [16] T. Poggio E. K. Miller D. J. Freedman, M. Riesenhuber. Categorical representation of visual stimuli in the primate prefrontal cortex. *Neuroscience*, 291(5502):312–316, January 2001.
- [17] R. C. Miall D. M. Wolpert and M. Kawato. Internal models in the cerebellum, 1998.
- [18] J. Peters D. Nguyen-Tuong. Model learning in robotics: A survey. *Cognitive Processing*, 12(4):319–340, 2011.
- [19] B. K. Shaw D. Sparks, W. B. Kristan. The role of population coding in the control of movement.
- [20] G. Dileep. *How the brain might work: A hierarchical and temporal model for learning and recognition*. PhD thesis, 2008.
- [21] K. Doya. What are the computations of the cerebellum, the basal ganglia, and the cerebral cortex? *Neural Networks*, 12:961–974, 1999.
- [22] K. Doya. Complementary roles of basal ganglia and cerebellum in learning and motor control. *Current Opinion in Neurobiology*, 10, 2000.
- [23] K. Doya. Reinforcement learning: Computational theory and biological mechanisms. *HFSP Journal*, 1(1):30–40, May 2007.
- [24] G. G. Moisen E. A. Freeman. A comparison of the performance of threshold criteria for binary classification in terms of predicted prevalence and kappa. *Ecological Modelling*, 217:48–58, May 2008.
- [25] S. Parodi E. Bottarelli. Un approccio per la valutazione della validità dei test diagnostici: le curve r.o.c. (receiver operating characteristic). *Ann. Fac. Medic. Vet. di Parma*, 23:49–68, 2003.
- [26] W. Bialek E. Schneidman and M. J. Berry II. Synergy, redundancy, and independence in population codes. *The Journal of Neuroscience*, December 2003.

- 
- [27] C. Yu G. Singh H. Lee, Y. Shen and A. Y. Ng. Quadruped robot obstacle negotiation via reinforcement learning. In *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2006.
- [28] C. B. Hart and S. F. Giszter. A neural basis for motor primitives in the spinal cord. *The Journal of Neuroscience*, 30(4):1322–1336, 2010.
- [29] D. O. Hebb. *The Organization of Behavior*. 1949.
- [30] R. Holley. How good can it get? analysing and improving ocr accuracy in large scale historic newspaper digitisation programs. *D-Lib Magazine*, 15(3/4), 2009.
- [31] J. C. Houk and S. P. Wise. Distributed modular architectures linking basal ganglia, cerebellum, and cerebral cortex: Their role in planning and controlling action. *Cerebral cortex*, 1995.
- [32] A. Hyvärinen. New approximations of differential entropy for independent component analysis and projection pursuit. *Advances in Neural Information Processing Systems*, Volume 10:273–279, 1998.
- [33] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, pages 10(3):626–634, 1999.
- [34] A. Hyvärinen and E. Oja. Independent component analysis: Algorithms and applications. *Neural Networks*, pages 13(4–5):411–430, 2000.
- [35] Y. P. Ivanenko, R. E. Poppele, and F. Lacquaniti. Five basic muscle activation patterns account for muscle activity during human locomotion. *The Journal of Physiology*, 556(1):267–282, 2004.
- [36] D. H. Ballard J. A. Feldman. Connectionist models and their properties. *Cognitive Science*, 6:205–254, 1982.
- [37] J. Peters J. Kober, D. Bagnell. Reinforcement learning in robotics: A survey. *International Journal of Robotics Research*, 32(11):1236–1272, 2013.
- [38] R.E. Kandel, J.H. Schwartz, and J.M. Thomas. *Principles of Neural Science Fourth Edition*. McGraw-Hill, 2000.
- [39] R. Kurzweil. *How to create a mind: the secret of thought revealed*.

- 
- [40] K. Lorenz. *Evolution and Modification of Behavior*. The University of Chicago Press, 1965.
- [41] S. Papert M. Minsky. *Perceptrons: an introduction to computational geometry*. 1969.
- [42] R. Manzotti. Towards artificial consciousness. *Newsletter on Philosophy and Computers*, pages 07(1):12–15, 2007.
- [43] R. Manzotti. No time, no wholes: A temporal and causal-oriented approach to the ontology of wholes. *Axiomathes*, pages 19:193–214, 2009.
- [44] R. Manzotti and V. Tagliasco. From “behaviour-based” robots to “motivations-based” robots. *Robotics and Autonomous Systems*, pages 51(2–3):175–190, 2005.
- [45] E .A. Murray and S. P. Wise. Interactions between orbital prefrontal cortex and amygdala: advanced cognition, learned responses and instinctive behaviors. *Current opinion in neurobiology*, pages 20(2):212–20, 2010.
- [46] F. Mutti. *Toward the integration of neural mechanisms and cognition in biologically inspired robot*. PhD thesis, Politecnico di Milano, 2012.
- [47] R. Nieuwenhuys, J. Voogd, and C. van Huijzen. *The Human Central Nervous System: A Synopsis and Atlas*. Steinkpoff, 2007.
- [48] A. Oursland, J. De Paula, and N. Mahmood. Case studies of independent component analysis. *Numerical Analysis of Linear Algebra*, 2004.
- [49] J. Kreiman P. Keating, A. Alwan and C. Esposito. Materials from the production and perception of linguistic voice quality project at ucla.
- [50] A. Papoulis. Probability, random variables, and stochastic processes (3rd ed.). 1991.
- [51] A. E. Patla. Some characteristics of emg patterns during locomotion: implications for the locomotor control process. *Journal of Motor Behavior*, 17:443–461, 1985.
- [52] A. E. Patla, T. W. Calvert, and R. B. Stein. Model of a pattern generator for locomotion in mammals. *AJP - Regulatory Integrative and comparative Physiology*, 248:R484–R494, 1985.

- 
- [53] D. M. W. Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.
- [54] Ryan Rifkin and Aldebaro Klautau. In defense of one-vs-all classification. *Journal of Machine Learning Research*, 5:101–141, 2004.
- [55] F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.
- [56] P. E. Latham S. Deneve and A. Pouget. Efficient computation and cue integration with noisy population codes. *Nature Neuroscience*, 18, 2001.
- [57] A. K. Ahlawat S. N. Mathur and V. P. Vishwakarma. Illumination invariant face recognition using supervised and unsupervised learning algorithms. *World Academy of Science, Engineering and Technology*, 19, 2008.
- [58] S. Schaal, J. Peters, J. Nakanishi, and A. J. Ijspeert. Learning movement primitives. *Springer Tracts in Advanced Robotics*, pages 561–572, 2005.
- [59] D.L. Schacter. *Psychology*.
- [60] S. Sherman and R. Guillery. *Exploring the Thalamus*. Elsevier, 2000.
- [61] O. Sporns. *Networks of the Brain*. The MIT Press, 2010.
- [62] R. S. Sutton and A. G. Barto. Reinforcement learning : Introduction, 1998.
- [63] F. M. Montes Gonzalez T. J. Prescott and al. A robot model of the basal ganglia: Behavior and intrinsic processing. *Neural networks*, 19:31–61, 2006.
- [64] G. Paynter T. Powell. Going grey? comparing the ocr accuracy levels of bitonal and greyscale images. *D-Lib Magazine*, 15(3/4), 2009.
- [65] S. J. Thorpe. Spike arrival times: A highly efficient coding scheme for neural networks. *Parallel processing in neural systems*, 1990.
- [66] S. J. Thorpe and M. Fabre-Thorpe. Seeking categories in the brain. *Neuroscience*, 291(5502):260–263, January 2001.

- [67] L. M. Ward. The thalamic dynamic core theory of conscious experience. *Consciousness and Cognition*, pages 20:464–486, 2011.



# Appendice A

## ICA

L'Analisi dei Componenti Indipendenti (ICA) è un algoritmo creato per estrarre dei segnali indipendenti da un gruppo di segnali dello stesso tipo mischiati insieme. Si basa sull'assunzione che ogni segnale può essere ricostruito basandosi sulla combinazione di un gruppo di segnali indipendenti, le Componenti Indipendenti [34, 48]. Questa analisi è usata spesso quando vogliamo sapere la loro forma originale di due o più segnali in ingresso uniti tra loro, andando quindi a separarli.

Supponiamo di avere quattro segnali generati dalle seguenti funzioni:

$$\sin\left(\frac{x}{2}\right) \quad (\text{A.1})$$

$$\left(\frac{((x \bmod 27) - 13)}{9}\right)^5 \quad (\text{A.2})$$

$$\frac{(x \bmod 27) - 13}{9} \quad (\text{A.3})$$

$$(x \times 2 - 1) \times \log\left(\frac{y}{10}\right) \begin{cases} 0 \leq x \leq 1 \\ 1 \leq y \leq 10 \end{cases} \quad (\text{A.4})$$

da queste funzioni sono stati ottenuti i quattro segnali che sono stati tracciati in Figura A.1a.

Sempre a partire dagli stessi segnali, ma modificandoli e facendone una media pesata con dei valori generati casualmente, abbiamo ottenuto i segnali mostrati in Figura A.1b. Questi rappresentano un esempio di segnali mischiati che possono presentarsi in ingresso ai sensori anche se sono originati da fonti diverse. La Figura A.1c mostra quattro nuovi segnali trovati dall'algoritmo delle componenti indipendenti a partire dai segnali di Figura A.1b. Come possiamo vedere, l'algoritmo è in grado di isolare le componenti indipendenti del segnale anche in presenza di rumore, e si può osservare come

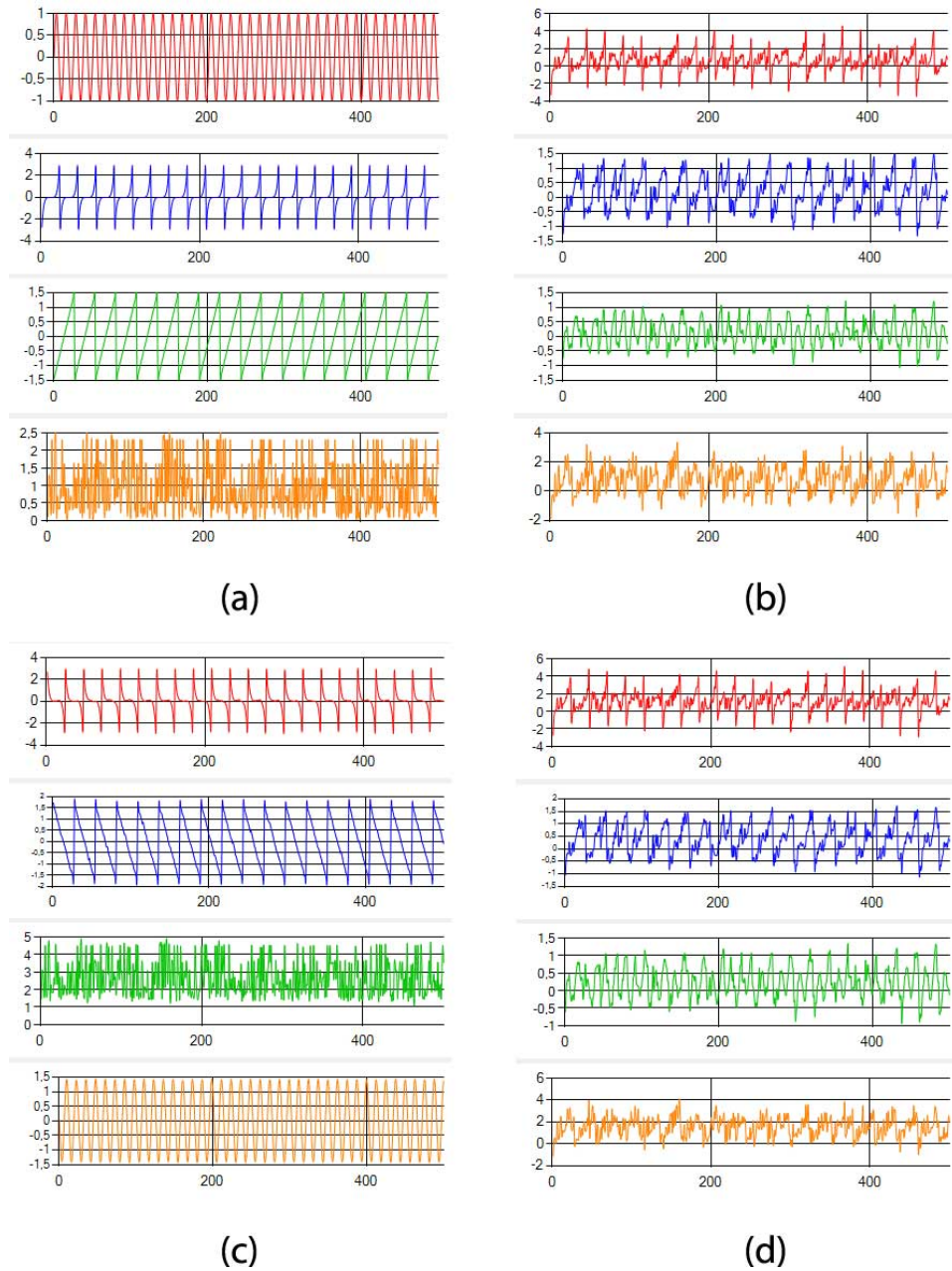


Figura A.1: ICA (a) segnali originali; (b) segnali mixati; (c) componenti indipendenti; (d) segnali ricostruiti.

questi corrispondano ai quattro segnali originali che avevamo mixato. Come spiegato nella sezione successiva, i componenti indipendenti sono ottenuti, a partire dai segnali in ingresso, applicando una determinata matrice dei pesi. Usando il processo inverso, invece, possiamo ottenere nuovamente i campioni originali. La Figura A.1d mostra i segnali ricostruiti usando ICA e la matrice dei pesi che abbiamo usato in precedenza.

Tuttavia, come possiamo osservare in Figura A.1c, l'algoritmo non è in grado di identificare la giusta ampiezza delle componenti indipendenti; infatti, mentre in Figura A.1a possiamo notare come l'ampiezza della sinusoide vari tra  $-1$  e  $+1$ , l'ampiezza corrispondente nel segnale indipendente varia tra  $-1.4$  e  $+1.4$ . Questo è un problema noto di ICA [34] e dipende dal fatto che, essendo sconosciute sia  $S$  che  $A$  nell'Equazione A.6, qualsiasi moltiplicatore applicato a  $s_i$  potrebbe essere annullato applicando lo stesso divisore ad  $a_i$ .

Per questo motivo, nella nostra implementazione del Modulo Intenzionale, abbiamo normalizzato i valori dei componenti indipendenti prima di applicarli ai segnali in ingresso; così facendo assumiamo che ogni componente indipendente abbia varianza  $E(s_i^2) = 1$ . Bisogna tener presente che questo non risolve il problema dell'ambiguità per quanto riguarda il segno: potremmo, infatti, avere ancora il segnale  $s_i$  moltiplicato per  $-1$ .

## A.1 Principi di funzionamento di ICA

Ogni segnale  $X_i$  in ingresso può essere visto come una combinazione lineare di componenti indipendenti:

$$X_i = \sum_{j=0}^N a_{ij} S_j \quad (\text{A.5})$$

dove  $S_j$  è un segnale delle componenti indipendenti e  $a_{ij}$  è un peso specifico da aggiungere a una componente in modo da poter ottenere il segnale in ingresso. Il sistema può essere descritto come

$$X = AS \quad (\text{A.6})$$

dove ogni riga di  $X$  è un segnale  $X_i$ , ogni colonna di  $S$  è un segnale delle componenti indipendenti  $S_i$  e  $A$  è una matrice  $n \times n$  che genera  $X$  a partire da  $S$ . Lo scopo di ICA è di trovare  $S$  e  $A$  conoscendo  $X$ .

Possiamo considerare ogni segnale come una variabile casuale. Vogliamo trovare una matrice

$$W = A^{-1} \quad (\text{A.7})$$

in cui

$$S = WX \quad (\text{A.8})$$

tale formula indica che ogni segnale indipendente  $S_i$  può essere espresso come una combinazione lineare dei segnali presenti in  $X$ . Bisogna tener presente che  $A$  deve essere invertibile affinché  $S = WX$  sia valida.

ICA si basa sull'ipotesi che ogni riga di  $S$  sia statisticamente indipendente; cioè, dati due componenti  $s_1$   $s_2$ , le informazioni dei valori di  $s_1$  non possono fornire nessuna informazione dei valori di  $s_2$  e viceversa. Questo può essere espresso dalla formula:

$$P(s_1|s_2) = P(s_1) \Rightarrow P(s_1, s_2) = \frac{P(s_1, s_2)}{P(s_2)} \Rightarrow P(s_1, s_2) = P(s_1) \times P(s_2) \quad (\text{A.9})$$

dove  $P(s_1, s_2)$  è la funzione di probabilità congiunta di  $s_1$  e  $s_2$ .  
Un'altra proprietà di variabili indipendenti casuali è

$$E(f_1(s_1), f_2(s_2)) = E(f_1(s_1))E(f_2(s_2)) \quad (\text{A.10})$$

per ogni funzione  $f_1, f_2$ .

Un'altra, seppur più debole, forma di indipendenza è la non correlazione. Due variabili  $s_1$  e  $s_2$  sono non correlate se la loro covarianza è 0:

$$Cov(s_1, s_2) = E(s_1 \times s_2) - E(s_1)E(s_2) = 0 \quad (\text{A.11})$$

Un'altra restrizione fondamentale presente in ICA è che la componente indipendente non può essere gaussiana. Il teorema del limite centrale afferma che la somma pesata di variabili casuali indipendenti tende verso una distribuzione gaussiana. Quindi,  $x_i = a_{i1}s_1 + a_{i2}s_2$  si avvicina ad una distribuzione di questo tipo molto più di  $s_1$  o  $s_2$ .

Questo implica che possiamo trovare la combinazione dei segnali indipendenti che formano  $X$  solo se non rappresentano già una distribuzione gaussiana. Questo può essere osservato nel grafico della distribuzione gaussiana, come mostrato in Figura A.2: la densità è completamente simmetrica, rendendo di fatto impossibile determinare qualsiasi informazione sulla direzione delle colonne nella matrice  $A$ . Per poter trovare i segnali indipendenti abbiamo bisogno di un modo per misurare la non gaussianità della matrice  $S$ . Semplificando il processo, manipoliamo i dati così da avere un segnale in ingresso  $X$  con media nulla e varianza pari a 1. In seguito, misuriamo la sua non gaussianità usando l'entropia negativa. L'entropia negativa si basa sulla quantità di informazione teorica dell'entropia differenziale. Per un vettore

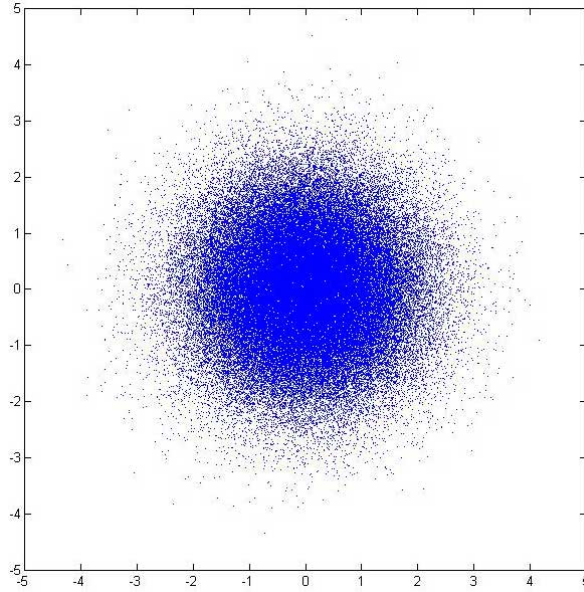


Figura A.2: Esempio di distribuzione gaussiana

casuale continuo  $y$  può essere definita come [15, 50]:

$$H(y) = - \int f(y) \log f(y) \quad (\text{A.12})$$

dove  $a_i$  è uno di tutti i possibili valori di  $Y$ . L'entropia negativa  $J$  può essere definita come:

$$J(y) = H(y_{gauss}) - H(y) \quad (\text{A.13})$$

in cui  $y_{gauss}$  è una variabile casuale con distribuzione gaussiana e matrice di covarianza della variabile  $y$ . L'entropia negativa è sempre maggiore o uguale a 0 e vale esattamente 0 se e solo se  $y$  ha una distribuzione di tipo gaussiano. Quindi l'entropia negativa è un ottimo stimatore statistico di non gaussianità. Noi utilizzeremo un'approssimazione dell'entropia negativa [32]:

$$J(y) \approx \sum_{i=1}^p k_i (E(G_i(y)) - E(G_i(v)))^2 \quad (\text{A.14})$$

in cui  $k_i$  sono costanti positive e  $v$  è una variabile gaussiana di media 0 e varianza 1, cioè una variabile standardizzata così come  $y$ .  $G_i$  sono alcune funzioni non quadratiche. La scelta di  $G$  è cruciale per stimatori robusti e per delle buone approssimazioni dell'entropia negativa; scelte appropriate sono [33]:

$$g_1(u) = \tanh(a_1 u) \quad , \quad g_2(u) = u e^{(-u^2/2)} \quad (\text{A.15})$$

dove  $1 \leq a_1 \leq 2$  è una costante, di solito equivalente a 1.

Nella nostra implementazione abbiamo usato la tangente iperbolica.

## A.2 L'algoritmo

Per i nostri scopi abbiamo implementato in MATLAB l'algoritmo FastICA presentato da A. Hyvärinen [33]. Questo algoritmo è composto da alcuni passi:

- *Preprocessing*: prima di applicare l'algoritmo di analisi delle componenti indipendenti al segnale in ingresso  $X$ , dobbiamo svolgere alcune azioni per semplificare la computazione:
  - *Centering*: assumiamo che ogni ingresso  $X_i$  ha come valore medio 0. Quindi, sottraiamo la media da ogni riga di  $X$ :

$$X_i = X_i - E(X_i) \quad (\text{A.16})$$

Bisogna notare che questo implica anche che  $S_i$  avrà media nulla, come si può notare nell'Equazione A.6.

- *Covariance*: calcoliamo la covarianza di  $X$ , che è:

$$\text{Cov}(X) = E(XX^T) \quad (\text{A.17})$$

La matrice della covarianza è quadrata e simmetrica. In seguito, possiamo calcolare il valore singolare di decomposizione (Singular Value Decomposition, SVD) dalla matrice di covarianza, in modo da ottenere la matrice diagonale degli autovalori  $D$  e la matrice ortogonale degli autovettori  $E$ :

$$\text{Cov}(X) = EDE^T \quad (\text{A.18})$$

- *Whitening*: trasformiamo ogni vettore  $x$  linearmente così da ottenere un nuovo vettore  $x_w$  che risulta sbiancato (whitened) così che i suoi componenti non siano correlati e che la loro varianza sia uguale a 1. Questo implica anche che la matrice di covarianza di  $X_w$  è la matrice identità:

$$\text{Cov}(X_w) = E(X_w X_w^T) = I \quad (\text{A.19})$$

Il whitening può essere ottenuto con:

$$X_w = ED^{-1/2}E^T X \quad (\text{A.20})$$

dove  $D^{-1/2} = \text{diag}(d_1^{-1/2}, \dots, d_n^{-1/2})$ . Il whitening trasforma i segnali mixati  $X$  in una nuova matrice,  $A_w$ ; la matrice originale può essere ottenuta dall'Equazione A.6 e A.20 con:

$$X_w = ED^{-1/2}E^T AS = A_w S \quad (\text{A.21})$$

La nuova matrice sbiancata  $A_w$  è ortogonale, quindi:

$$\text{Cov}(X_w) = E(X_w X_w^T) = A_w E(SS^T) A_w^T = A_w A_w^T = I \quad (\text{A.22})$$

In aggiunta, il processo di whitening riduce il numero di parametri che dobbiamo stimare.

- *L'algoritmo FastICA*: per ogni componente indipendente che vogliamo trovare definiamo un vettore di pesi  $w$  (da notare che questo implica che il numero di componenti indipendenti deve essere conosciuto a priori dall'algoritmo; cambiare il numero di componenti indipendenti può portare a risultati differenti. Assumiamo quindi che il numero di componenti indipendenti sia uguale al numero di segnali in ingresso). La regola di apprendimento dell'algoritmo trova una direzione, che converge ad un vettore  $w$  così che  $s = w^T x$  massimizza la non gaussianità.

Il ciclo principale dell'algoritmo è:

1. Crea una matrice di pesi con valori casuali  $W$ , delle stesse dimensioni di  $A_w$ ;
2. Poiché abbiamo segnali differenti, dobbiamo definire un vettore  $w_i$  per ogni riga di  $S$ . In aggiunta dobbiamo assicurarci che diversi vettori non convergano allo stesso massimo; dobbiamo cioè decorrelare ogni uscita  $w_i^T x_i$  ad ogni iterazione.

Questo può essere ottenuto usando la decorrelazione di Gram-Schmidt:

$$w_{ij} = \frac{w_{ij}}{\sqrt{\sum_{i=0}^n w_{ij}^2}} \quad \forall \text{column } j = 0, \dots, n$$

$$w_{ik} = w_{ik} - (w_{ik} \times \sum_{i=0}^n w_{ik}^2) \quad \forall \text{column } k = j + 1, \dots, n$$
(A.23)

3. Controllo della convergenza tramite:

$$\begin{aligned} \text{minAbsCos} &= \text{min}(\text{abs}(\text{diag}(W^T W_{old}))); \\ \text{if}(1 - \text{minAbsCos} < \text{epsilon}) \\ &\text{break}; \end{aligned} \quad (\text{A.24})$$

in cui  $W_{old}$  è il valore di  $W$  al passo precedente, e  $\epsilon$  è un valore predefinito.

4. Aggiornamento dei pesi di  $W$ : ortogonalizziamo  $W$  tenendo presenti i componenti precedenti:

$$W = E(X G(X^T W)) - E(G'(W^T X))W \quad (\text{A.25})$$

quindi normalizziamo  $W$

$$W = W/\|W\| \quad (\text{A.26})$$

5. Tornare al punto 1.
6.  $S = WX_w$  ;

### A.3 Conclusioni

L'implementazione di questo algoritmo mostra alcune proprietà interessanti [33]:

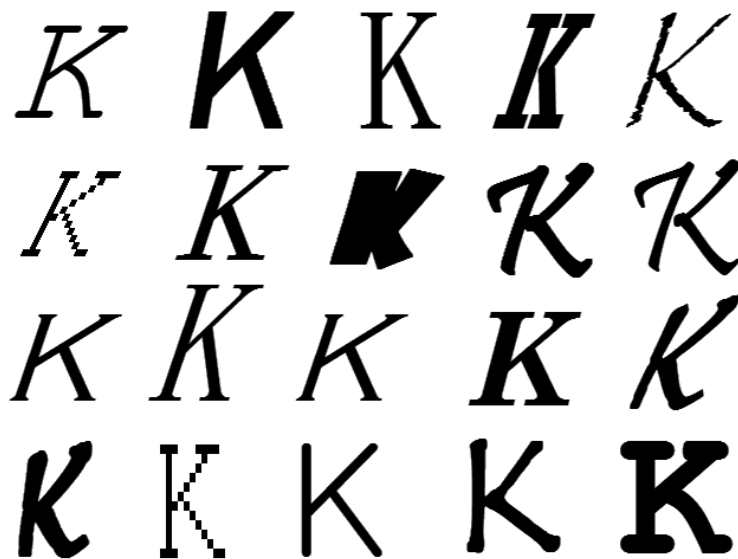
- La convergenza è cubica o, per lo meno, quadratica, mentre altre implementazioni di ICA basate sulla discesa del gradiente presentano una convergenza lineare. Quindi la convergenza illustrata è molto più veloce.
- Differentemente dall'algoritmo basato sulla discesa del gradiente, non è necessario che si indichi di quanto variare il passo, rendendo l'algoritmo più semplice.
- Il metodo può migliorare le proprie prestazioni scegliendo una valida funzione non lineare  $g$ . Le funzioni mostrate nell'Equazione A.15 sono buoni esempi di funzioni ottimali per robustezza e varianza minima.
- L'algoritmo è capace di trovare ogni componente indipendente di ogni distribuzione non gaussiana usando qualsiasi funzione non lineare  $G$ , a differenza di altri algoritmi che necessitano di una stima della distribuzione probabilistica e una relativa funzione di non linearità.
- Presenta molti vantaggi delle reti neurali: è parallelo, distribuito e poco costoso sia da un punto di vista computazionale, sia da un punto di vista di utilizzo della memoria.



## Appendice B

### Esempi Dataset

In questa appendice presentiamo le immagini dei dataset utilizzati per i due esperimenti visivi illustrati nel Capitolo 6. Nelle Figure B.1 e B.2 mostriamo rispettivamente alcuni esempi dell'insieme usato per l'addestramento e alcuni relativi a quello per il test.



*Figura B.1: Esempio di immagini di addestramento per la singola lettera.*

Nelle Figure B.4 e B.3 mostriamo, invece, rispettivamente un insieme di immagini di addestramento e i volti per il test sugli stessi volti.

Nelle figure presentate in Figura B.5, è possibile vedere le immagini utilizzate per un singolo soggetto del campione, tratte dal database B di Yale [1].



Figura B.2: Esempio di immagini di test per la singola lettera.



Figura B.3: Esempio di immagini di test per un singolo soggetto.



*Figura B.4: Esempio di immagini di addestramento per un singolo soggetto.*



*Figura B.5: Esempio di immagini presenti nel database, in particolare quelle appartenenti al primo soggetto preso in considerazione.*