
Reconhecimento visual de gestos para imitação
e correção de movimentos em fisioterapia
guiada por robô

Ricardo Fibe Gambirasio

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Ricardo Fibe Gambirasio

Reconhecimento visual de gestos para imitação e correção de movimentos em fisioterapia guiada por robô

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências - Ciências de Computação e Matemática Computacional. VERSÃO REVISADA

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientadora: Prof^a. Dr^a. Roseli Aparecida Francelin Romero

USP - São Carlos
Janeiro de 2016

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados fornecidos pelo(a) autor(a)

G191r Gambirasio, Ricardo Fibe
Reconhecimento visual de gestos para imitação e
correção de movimentos em fisioterapia guiada por
robô / Ricardo Fibe Gambirasio; orientadora Roseli
Aparecida Francelin Romero. -- São Carlos, 2016.
70 p.

Dissertação (Mestrado - Programa de Pós-Graduação
em Ciências de Computação e Matemática
Computacional) -- Instituto de Ciências Matemáticas
e de Computação, Universidade de São Paulo, 2016.

1. Reconhecimento de gestos. 2. Fisioterapia. 3.
Visão robótica. 4. Interação humano-robô. 5. Máquinas
de estados finitos. I. Romero, Roseli Aparecida
Francelin, orient. II. Título.

Ricardo Fibe Gambirasio

Visual gesture recognition for mimicking and correcting movements in robot-guided physiotherapy

Master dissertation submitted to the Instituto de Ciências Matemáticas e de Computação - ICMC-USP, in partial fulfillment of the requirements for the degree of the Master Program in Computer Science and Computational Mathematics. FINAL VERSION

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. Roseli Aparecida Francelin Romero

USP - São Carlos
January 2016

Resumo

O objetivo deste trabalho é tornar possível a inserção de um robô humanoide para auxiliar pacientes em sessões de fisioterapia. Um sistema robótico é proposto que utiliza um robô humanoide, denominado NAO, visando analisar os movimentos feitos pelos pacientes e corrigi-los se necessário, além de motivá-los durante uma sessão de fisioterapia.

O sistema desenvolvido permite que o robô, em primeiro lugar, aprenda um exercício correto de fisioterapia observando sua execução por um fisioterapeuta; em segundo lugar, que ele demonstre o exercício para que um paciente possa imitá-lo; e, finalmente, corrija erros cometidos pelo paciente durante a execução do exercício.

O exercício correto é capturado por um sensor *Kinect* e dividido em uma sequência de estados em dimensão espaço-temporal usando *k-means clustering*. Estes estados então formam uma máquina de estados finitos para verificar se os movimentos do paciente estão corretos. A transição de um estado para o próximo corresponde a movimentos parciais que compõem o movimento aprendido, e acontece somente quando o robô observa o mesmo movimento parcial executado corretamente pelo paciente; caso contrário o robô sugere uma correção e pede que o paciente tente novamente.

O sistema foi testado com vários pacientes em tratamento fisioterapêutico para problemas motores. Os resultados obtidos, em termos de precisão e recuperação para cada movimento, mostraram-se muito promissores. Além disso, o estado emocional dos pacientes foi também avaliado por meio de um questionário aplicado antes e depois do tratamento e durante o tratamento com um *software* de reconhecimento facial de emoções e os resultados indicam um impacto emocional bastante positivo e que pode vir a auxiliar pacientes durante tratamento fisioterapêuticos.

Palavras-chave: Reconhecimento de gestos, Fisioterapia, Visão robótica, Interação humano-robô, Máquinas de estados finitos.

Abstract

This dissertation develops a robotic system to guide patients through physiotherapy sessions. The proposed system uses the humanoid robot NAO, and it analyses patients' movements to guide, correct, and motivate them during a session.

Firstly, the system learns a correct physiotherapy exercise by observing a physiotherapist perform it; secondly, it demonstrates the exercise so that the patient can reproduce it; and finally, it corrects any mistakes that the patient might make during the exercise.

The correct exercise is captured via Kinect sensor and divided into a sequence of states in spatial-temporal dimension using k-means clustering. Those states compose a finite state machine that is used to verify whether the patient's movements are correct. The transition from one state to the next corresponds to partial movements that compose the learned exercise. If the patient executes the partial movement incorrectly, the system suggests a correction and returns to the same state, asking that the patient try again.

The system was tested with multiple patients undergoing physiotherapeutic treatment for motor impairments. Based on the results obtained, the system achieved high precision and recall across all partial movements. The emotional impact of treatment on patients was also measured, via before and after questionnaires and via a software that recognizes emotions from video taken during treatment, showing a positive impact that could help motivate physiotherapy patients, improving their motivation and recovery.

Key-words: Gesture recognition, Physiotherapy, Robotic vision, Human-robot interaction, Finite state machines.

Sumário

Lista de Figuras	13
Lista de Tabelas	15
1 Introdução	17
1.1 Justificativa	19
1.2 Objetivos	21
1.3 Organização	21
2 Trabalhos relacionados	23
2.1 Robôs na medicina	23
2.2 Reconhecimento, imitação e interpretação de gestos	25
2.2.1 Algoritmo de Viterbi	27
2.2.2 Algoritmo de Baum-Welch	29
2.2.3 Máquinas de estados finitos	30
2.3 Teleoperação	31
3 Materiais e metodologia	35
3.1 Robô NAO	35
3.2 Sensor Kinect	38
3.3 Imitação de gestos	39
3.3.1 Fundamentos	39
3.3.2 Mapeando articulações	40
3.4 Algoritmo k-means	42
3.5 Reconhecimento de emoções via modelagem geométrica de faces	43
3.6 Considerações Finais	46
4 Método Proposto	47
4.1 Modelando gestos via máquinas de estados finitos	47
4.1.1 Construção da Máquina de Estados Finitos	48
4.2 Sistema Integrado para Interação com o Paciente	51
4.3 Avaliação do Sistema Integrado	52

4.3.1	Avaliação do algoritmo de reconhecimento de gestos	53
4.3.2	Avaliação do estado emocional dos pacientes	55
4.3.3	Reconhecimento de emoções via modelamento geométrico de faces .	55
4.4	Considerações Finais	57
5	Conclusão	59
5.1	Trabalhos futuros	61
	Referências	63

Lista de Figuras

2.1	NAO interage com crianças autistas. Fonte: Ubifrance	24
2.2	Rede neural identifica ângulos correspondentes à base de dados para reconhecimento de gestos, em Waldherr, Romero, and Thrun (2000)	27
2.3	Modelo oculto de Markov Fonte: Wikipedia	28
2.4	Algoritmo de Viterbi Fonte: Wikipedia	29
3.1	Componentes do robô NAO Fonte: Aldebaran Robotics	36
3.2	API para fala do robô NAO Fonte: Aldebaran Robotics	36
3.3	API para movimentação do robô NAO Fonte: Aldebaran Robotics	37
3.4	Componentes do Kinect Fonte: Microsoft	38
3.5	Junções e ângulos considerados na imagem Fonte:Zuher and Romero (2012)	42
3.6	Algoritmo k-means identificando agrupamentos distintos Fonte: Wikipedia	43
3.7	Pontos considerados para análise de emoções Fonte:Libralon and Romero (2014)	45
4.1	Kinect Skeleton Fonte: Microsoft	48
4.2	Robô NAO guia paciente em sessão de fisioterapia	50
4.3	Interação entre fisioterapeuta, robô, e paciente para um exercício	51
4.4	Impacto no estado emocional do paciente, comparando-se respostas ao questionário antes e depois do tratamento.	55
4.5	Rostos de pacientes mapeados pelo sistema de Libralon and Romero (2013), mostrando emoções de alegria e neutralidade.	56
4.6	Impacto no estado emocional do paciente, comparando-se respostas ao questionário antes e depois do tratamento.	56

Lista de Tabelas

4.1	Matriz de confusão com resultados da avaliação do algoritmo	54
-----	---	----

Introdução

De robôs cirurgiões teleoperados a minúsculos robôs que removem placa arterial, sistemas robóticos têm assumido um papel cada vez mais importante na medicina. Nos próximos anos, espera-se que milhares de robôs passem a atuar no setor de saúde, reduzindo custos e facilitando o acesso para pessoas mais pobres, sem o risco de erro humano causado por distrações ou fadiga (McNickle (2012)). Entre estes robôs, o humanoide NAO, de aparência simpática e amigável, é extremamente popular com crianças (e com muitos adultos também), e vem sendo utilizado com sucesso em tratamentos experimentais de crianças com necessidades especiais (Guldberg (2013)).

Em particular, o campo de reabilitação robótica dedica-se ao uso de equipamentos robóticos para entender e melhorar a reabilitação de pacientes, desenvolvendo tecnologias que possam ser usadas com facilidade por pacientes, médicos e terapeutas, para melhorar o dia-a-dia dos pacientes e tornar mais efetivos seus tratamentos. Este campo inclui cadeiras de rodas robóticas e outros equipamentos para auxílio à mobilidade, robôs companheiros, manipuladores de membros para pessoas com restrições motoras e robôs educativos. Os robôs para reabilitação podem ser usados em diversos ambientes, como hospitais, centros de fisioterapia, escolas e, eventualmente, até na casa do paciente (Matarić, Tapus, Winstein, and Eriksson (2009)).

Inclui-se aí o desenvolvimento de equipamentos específicos para auxiliar o uso de diferentes faculdades sensoriais e motoras (em braços, pernas etc), para auxiliar o treinamento terapêutico e para avaliar a performance e o progresso senso-motor do paciente (Marchal-Crespo and Reinkensmeyer (2009)). O uso de equipamentos robóticos para fins de reabilitação em geral é bem recebido por pacientes e tem se provado efetivo como terapia auxiliar para indivíduos sofrendo de desabilidades motoras provindas, por exemplo, de derrames (Brewer, McDowell, and Worthen-Chaudhari (2007)).

A reabilitação robótica vem se desenvolvendo desde a década de 60, com a construção

do manipulador CASE, um suporte para braço controlado automaticamente que movia com quatro graus de liberdade o braço paralisado de um paciente. Outro exemplo desta época é o manipulador Rancho Los Amigos, este com sete graus de liberdade (Kim and Cook (1985)). Na década de 70, a reabilitação robótica evoluiu e se tornou mais especializada, com o desenvolvimento de sistemas para assistência no trabalho de escritório, auxiliando a manipulação de telefones, cadernos e computadores, usando comandos pré-programados (Roesler, Kuppers, and Schmalenbach (1978)). No Centro Prostético para Veteranos em Nova York, foi criado o primeiro braço robótico acoplado a uma cadeira de rodas, com mecanismo telescópico que permitia alcançar o chão ou o teto (Mason and Peizer (1978)). O *Unimation Puma* foi um robô de uso industrial frequentemente adaptado para uso em reabilitação, com o encorajamento de seu fabricante. Um uso pioneiro deste robô foi desenvolvido pela Boeing em Seattle, inicialmente para uso por seus próprios programadores com deficiências (Fu (1986)).

Em geral, a maneira mais barata e eficiente para produzir um sistema robótico de reabilitação é adaptar um robô já disponível comercialmente. Mas existem boas razões para produzir um robô específico para este fim, como uma melhor integração do sistema como um todo, maior eficiência no domínio desejado, redução de custo dependendo dos requisitos do sistema e independência de um fornecedor que pode não compartilhar dos mesmos objetivos (Hillman (2003)). Muitas vezes sistemas desenvolvidos especificamente para pesquisa em reabilitação são posteriormente produzidos comercialmente, como no caso do robô *Regenesis*, desenvolvido pela *Neil Squire Foundation* no Canadá (Fengler and Cameron (1988)).

Também foram desenvolvidos robôs para auxiliar na alimentação de pessoas com impedimentos motores. A empresa *Rehab Robotics* teve início em 1987 como um projeto de mestrado de seu fundador, para ajudar seu vizinho de 12 anos com paralisia espástica a se alimentar, usando um robô educacional da *Cyber Robotics* (Topping (2001)). Hoje, a *Rehab Robotics* é uma das empresas que mais vende robôs para reabilitação, capazes de auxiliar em tarefas como escrita, fazer a barba e aplicação de maquiagem.

A tecnologia de veículos auto-guiados também foi aplicada à reabilitação, como o robô *Meldog* (Tachi, Tanie, Komoriya, and Abe (1985)), construído para guiar deficientes visuais em um ambiente mapeado, substituindo um cão guia, e várias cadeiras de rodas "inteligentes", capazes de desviar de obstáculos, seguir corredores e manobrar para passar por portas (Nisbet, Odor, and Loudon (1988); Borgolte, Hoelper, and Hoyer (1997)).

Aplicações terapêuticas da tecnologia robótica para reabilitação vêm ganhando espaço mais recentemente. O *MIME system* (Shor, Lum, Burgar, Van der Loos, Majmundar, and Yap (2001)) é usado para reabilitação de pacientes que sofreram derrames cerebrais, com dois braços robóticos acoplados aos braços do paciente e sendo movidos de forma ativa, passiva ou bilateral. O projeto *GENTLE/S* (Amirabdollahian, Loureiro, Driessen, and Harwin (2001)) requer que o paciente se mova empurrando um braço robótico com *feedback* háptico em um ambiente gerado virtual tri-dimensional. Khalili et. al. [Khalili

and Zomlefer (1988)] usaram dois robôs planares para manipular juntas da perna após cirurgia, com um robô segurando a parte inferior da perna e outro a parte superior, com as juntas em compressão para melhorar a recuperação pós-cirúrgica.

1.1 *Justificativa*

Apesar do crescente sucesso obtido com a utilização de sistemas robóticos na medicina nos mais diversos contextos, inclusive para a reabilitação de pacientes com dificuldades motoras providas de problemas de saúde como derrames cerebrais, ainda está em aberto a questão de como usar robôs de forma eficaz e produtiva para guiar sessões de fisioterapia.

Há um crescente interesse no uso de sistemas robóticos para a reabilitação motora em pacientes com dificuldades motoras causadas por danos neurológicos como derrames cerebrais e danos à espinha dorsal (Marchal-Crespo and Reinkensmeyer (2009)). Derrames cerebrais são uma das principais causas de deficiências neurológicas no mundo, com cerca de 795.000 pessoas sofrendo derrames por ano somente nos Estados Unidos e a maioria dos afetados acaba com alguma dificuldade motora (Mozaffarian, Benjamin, Go, Arnett, Blaha, Cushman, de Ferranti, Despres, Fullerton, Howard, et al. (2015)). Mas, com o uso adequado de treinamento para os membros afetados durante a fase crítica pós-derrame, estas dificuldades motoras podem ser significativamente reduzidas (Taub, Uswatte, King, Morris, Crago, and Chatterjee (2006)).

A qualidade da recuperação depende fortemente da quantidade de exercícios terapêuticos realizados, bem como, da severidade do derrame e da capacidade cognitiva do paciente. Teasell et. al. [Teasell and Kalra (2005)] apresentaram ampla evidência de que a intensidade e a frequência da terapia focada contribuem para melhorar o resultado final na mobilidade do paciente. Entretanto, como este tipo de tratamento normalmente requer supervisão por profissionais treinados, a falta de recursos limita o tratamento que muitos pacientes acabam recebendo, reduzindo dramaticamente a qualidade de vida do paciente, estendendo o tempo e custos de tratamento médico e também perdendo produtividade na atividade econômica do paciente (Matarić et al. (2009)).

Dada a alta incidência de derrames, além de seu potencial aumento com o envelhecimento da população global, a reabilitação robótica pós-derrame é uma área ativa de pesquisa. O paradigma mais comum é o uso de um equipamento robótico para manipulação dos membros do paciente durante o movimento de reabilitação, embora também se tenha usado robôs sem contato físico para auxiliar o paciente no seu tratamento.

Mataric [Matarić et al. (2009)] propôs um sistema robótico autônomo que monitora, encoraja e provê lembretes aos pacientes para que realizem seus exercícios. Kristy et al. [Kristy, Wu, Erlandson, deBear, Geer, and Dijkers (1989)] construíram um sistema para exercício composto por um braço robótico controlado por um computador com software que provê rotinas de exercícios e coleta dados sobre o paciente, permitindo que este realize alguns exercícios sem a supervisão imediata do terapeuta, terapeuta este que pode depois

ler o relatório quantitativo gerado pelo sistema para acompanhar o progresso do paciente. Lum et al. [Lum, Uswatte, Taub, Hardin, and Mark (2006)] desenvolveram um sistema de teleoperação para tratamento induzido por restrição onde o terapeuta controla remotamente as instruções fornecidas por um computador, obtendo resultados semelhantes aos de tratamentos presenciais convencionais.

Pode não ser óbvio por que a presença física do robô é necessária para guiar o paciente, já que as instruções poderiam ser passadas ao paciente em uma tela de computador. Mas há ampla evidência (Matarić et al. (2009)) da importância do robô presente junto ao paciente, com seu impacto positivo no engajamento e motivação do paciente. Para tanto, é importante que a personalidade do robô seja agradável e transmita um certo "calor humano" (Goetz and Kiesler (2002)), e possivelmente varie de acordo com a personalidade e humor do paciente em diferentes momentos.

Levando-se em conta o sucesso do robô NAO na interação com crianças em várias situações, ele se mostra um ótimo candidato para orientar pacientes em sessões de fisioterapia. Lange [Lange, Koenig, McConnell, Chang, Juang, Suma, Bolas, and Rizzo (2012)] obteve bons resultados com o uso de jogos de realidade virtual especialmente desenhados para tratamento fisioterápico de crianças com dificuldades motoras. O uso de video games diminui o absenteísmo (já que as crianças insistem para que os pais as levem para a próxima sessão), aumenta a motivação e concentração dos pacientes e gera progresso mais rapidamente do que tratamentos convencionais.

O uso do NAO para guiar sessões de fisioterapia pode ter efeito similar ao do uso de video games, diminuindo o absenteísmo e aumentando a motivação e a concentração das crianças, levando a resultados melhores e mais rápidos do que tratamentos convencionais.

Um sistema robótico capaz de guiar uma sessão de fisioterapia também pode levar a uma redução significativa do custo do tratamento, aumentando assim sua disponibilidade para pessoas com menos recursos. Embora o uso do robô não dispense totalmente a presença de um fisioterapeuta, ele permite que o fisioterapeuta supervisione o tratamento de um número maior de pacientes concomitantemente.

A literatura existente sobre o uso de robôs para guiar sessões de fisioterapia ainda é um pouco escassa, mas a evidência existente sugere que tais robôs são bem recebidos e têm impacto positivo no engajamento dos pacientes e no resultado final do tratamento, efeito este que é mais notável quando um robô humanoide com personalidade "humana" está presente fisicamente (Matarić et al. (2009)).

1.2 *Objetivos*

O objetivo deste Mestrado é o desenvolvimento de um sistema que permita que um robô humanoide oriente crianças em uma sessão de fisioterapia. O robô deve perceber — via sensores — e avaliar os movimentos da criança — via algoritmos estatísticos que levem em conta a ordem temporal de eventos para reconhecimento de gestos. O robô deve também prover *feedback*, comunicando acertos e erros e demonstrando os movimentos corretos, até que a criança realize os exercícios determinados pelo fisioterapeuta de forma correta.

Para uma melhor adaptação do paciente, é importante que os sensores utilizados sejam não-intrusivos, isto é, não necessitem que nenhum objeto seja colocado sobre o corpo do paciente e obtenham informações sem tocá-lo.

A experiência deve ser agradável para o paciente, o que contribui para seu engajamento durante a sessão e para motivá-lo a continuar o tratamento.

1.3 *Organização*

Este trabalho está organizado do seguinte modo. No Capítulo 2, são apresentados alguns dos principais trabalhos relacionados ao tema abordado nesta dissertação. No Capítulo 3, uma breve descrição do robô humanoide e do sensor utilizado são apresentadas, seguida pela técnica de imitação de gestos utilizada. No Capítulo 4, um método para modelamento de gestos é proposto e a avaliação do sistema integrado é apresentada. Finalmente, no Capítulo 5, a conclusão e os trabalhos futuros são apresentados.

Trabalhos relacionados

2.1 *Robôs na medicina*

Sistemas robóticos têm assumido um papel cada vez mais importante na medicina. O *da Vinci Surgical System*, aprovado em 2000 pela agência reguladora americana FDA, já realizou mais de 20.000 cirurgias. *Magnetic Microbots* são um grupo de minúsculos robôs que removem placa de artérias e auxiliam diagnósticos. O *Bestic Arm* auxilia pacientes com dificuldades motoras a comer. A Toyota está testando quatro modelos de robôs que auxiliam idosos a se equilibrar e caminhar. A *Vasteras Giraff* é um robô de controle remoto com câmera e videoconferência para facilitar a comunicação de pessoas com dificuldades motoras. O *Aethon TUG*, um robô para transporte de medicamentos e acessórios hospitalares, faz trabalho equivalente a três funcionários em tempo integral, custando menos que um único funcionário. Nos próximos anos, espera-se que milhares de robôs passem a atuar no setor de saúde, reduzindo custos e facilitando o acesso para pessoas mais pobres, sem o risco de erro humano causado por distrações ou fadiga (McNickle (2012)).

Para o tratamento de crianças, um aspecto fundamental é que a aparência e comportamento do robô sejam amigáveis e não as intimidem. O robô humanoide NAO, da empresa francesa *Aldebaran Robotics*, do tamanho de uma criança pequena (58 cm) e com movimentos articulados muito semelhantes a de um ser humano, é extremamente popular com crianças (e com muitos adultos também), e vem sendo utilizado com sucesso em tratamentos experimentais. A *Aldebaran Robotics* criou o projeto ASK NAO (*Autism Solutions for Kids*) para auxiliar professores, pais e médicos no tratamento de e comunicação com crianças autistas. Pesquisadores da Universidade de Birmingham observaram crianças autistas, incapazes de se comunicar ou fazer contato visual com humanos, se

comunicando com o NAO. Os pesquisadores atribuem este resultado ao fato de o robô não mostrar emoções, e portanto ser mais previsível e menos intimidador do que um ser humano (Guldborg (2013)).



Figura 2.1: NAO interage com crianças autistas.

Fonte: Ubifrance

Shamsuddin, Yussof, Ismail, Hanapiah, Mohamed, Piah, and Zahari (2012) também estão investigando os efeitos da interação entre crianças autistas e o NAO. Barakova and Lourens (2010) desenvolvem jogos que utilizam o NAO para ajudar crianças autistas a desenvolver habilidades sociais.

O NAO não vem sendo utilizado somente para o tratamento de crianças autistas. Ros, Nalin, Wood, Baxter, Looije, Demiris, Belpaeme, Giusti, and Pozzi (2011) realizaram uma série de experimentos com robôs na ala pediátrica de um hospital para avaliar as diferenças com relação ao ambiente controlado de um laboratório.

Gelin, daAlessandro, Le, Deroo, Doukhan, Martin, Pelachaud, Rilliard, and Rosset (2010) utilizam o NAO para contar e interpretar histórias para crianças. Delaborde, Tahon, Barras, and Devillers (2009) medem as reações de crianças durante a interação com o NAO para o desenvolvimento de um sistema de assistência a pessoas com dificuldades motoras.

Klein, Gaedt, and Cook (2013) exploram experiências com o robô emocional PARO (uma foca de pelúcia robótica) para propósitos terapêuticos, com foco em estímulos sensoriais para pessoas com mais de 80 anos, para reduzir seu isolamento social, aumentando seu bem-estar e reduzindo a incidência de demência.

Beran, Ramirez-Serrano, Vanderkooi, and Kuhn (2013) colocam crianças com medo de injeções para interagir com o robô NAO enquanto uma vacina lhes é aplicada. Cinquenta e sete crianças com idades entre quatro e nove anos foram separadas em dois grupos; um grupo foi vacinado da forma usual e o outro grupo interagiu com o robô durante a vacina. As crianças, pais, enfermeiras e pesquisadores avaliaram então a dor e ansiedade

da criança durante o procedimento, e o grupo que interagiu com o robô teve resultados bem melhores do que o grupo de controle. Além disso, as crianças que interagiram com o robô se recuperaram mais rapidamente, sorrindo e relaxando quase que imediatamente após a injeção, enquanto que as crianças no grupo de controle continuaram tristes e sem falar com os pais ou enfermeiras. A diferença foi tão marcante que os pesquisadores acabaram levando estas crianças para ver o robô após o experimento para tentar animá-las.

Krebs, Krams, Agrafiotis, DiBernardo, Chavez, Littman, Yang, Byttebier, Dipietro, Rykman, et al. (2014) utilizam a alta resolução com que sistemas robóticos conseguem registrar movimentos humanos para coletar medidas longitudinais de pacientes que sofreram derrames cerebrais. Os resultados mostram que o sistema robótico é capaz de detectar melhoras no performance motora dos pacientes com maior precisão e eficiência do que medidas clínicas tradicionais.

Cifuentes, Braidot, Rodríguez, Frisoli, Santiago, and Frizera (2012) desenvolvem uma ferramenta com sensores sem fio que medem a atividade cinemática e elétrica nos músculos de pacientes em reabilitação pós-derrame.

Gama, Chaves, Figueiredo, and Teichrieb (2012) desenvolvem um sistema de avaliação de movimentos de reabilitação usando o sensor Kinect para medir a posição do esqueleto e articulações do paciente, dando ao paciente *feedback* e produzindo uma nota para o movimento. O objetivo do sistema é seu uso em ambientes interativos de realidade virtual. Chaves, Figueiredo, Gama, de Araujo, and Teichrieb (2012) utilizam uma câmera RGB-D para reconhecer movimentos humanos em tempo real sem treinamento *offline*, baseando-se em pontos de referência, o que permite também identificar se o movimento está no começo, meio, ou fim. Da, Chaves, Marques-Oliveira, and Araújo (2012) desenvolvem um jogo fisioterapêutico controlado por movimentos de pacientes em reabilitação detectados pelo Kinect. Em um ambiente de realidade virtual, o paciente faz movimentos terapêuticos determinados por um fisioterapeuta para progredir no jogo, e recebe *feedback* em tempo real sobre sua performance.

2.2 Reconhecimento, imitação e interpretação de gestos

Gestos são dinâmicos, compostos por uma sequência de poses estáticas. Um mesmo gesto pode ser realizado em velocidades diferentes, levando à captura de poses diferentes. Um algoritmo de reconhecimento de gestos precisa ser capaz de procurar padrões nas dimensões espacial (em cada pose) e temporal (na ordem e velocidade das várias poses) (Lee and Kim (1999)). Algumas das principais técnicas para reconhecer gestos são *Dynamic Time Warping*, redes neurais, modelos ocultos de Markov, e máquinas de estados finitos.

Nishimura and Oka (1996) utilizam *Dynamic Time Warping* para reconhecer gestos em características extraídas de imagens em movimento. O algoritmo de *Dynamic Time Warping* mede a similaridade entre séries temporais com diferentes velocidades, calculando o *matching* ótimo entre as duas séries, “entortando-as” na dimensão temporal. O trabalho

obtem uma taxa reconhecimento de 80% para oito tipos de gestos.

Fels and Hinton (1993) usam redes neurais multicamadas para implementar um sistema de reconhecimento de gestos manuais capturados por uma *DataGlove* para produzir fala sintética. Com 7822 exemplos de treinamento, o sistema reproduz um vocabulário de 203 gestos, com menos de 1% de palavras erradas produzidas e 5% de gestos não reconhecidos.

Yamato, Ohya, and Ishii (1992), Yang and Xu (1994) e Lee and Kim (1999) utilizam modelos ocultos de Markov para reconhecer gestos. Cada estado oculto corresponde a uma pose de um possível gesto, e as poses observadas são consequências do estado oculto desconhecido. O algoritmo de Baum-Welch calcula as probabilidades de transição de estados do modelo que maximizam a probabilidade de se observar o conjunto de dados de treinamento. Finalmente, o algoritmo de Viterbi é utilizado para calcular a sequência de estados ocultos com maior probabilidade de gerar a sequência de poses observado. Esta sequência de estados ocultos de maior probabilidade corresponde ao gesto reconhecido. Este processo é explicado em detalhe mais adiante.

Um dos robôs pioneiros no reconhecimento, interpretação e interação com o ambiente foi o robô Shakey, desenvolvido entre 1966 e 1972 em Stanford. Shakey era capaz de reconhecer alguns gestos, navegar um ambiente com corredores e portas, abrir portas e acender luzes, e empurrar objetos. Alguns algoritmos famosos foram desenvolvidos para este projeto, como o A* e o grafo de visibilidade (Nilsson (1984)).

Mais recentemente, Waldherr et al. (2000) desenvolveram um sistema para controle de um robô móvel de maneira mais natural por um ser humano. Através de uma câmera, o robô segue a pessoa (filtrando a cor da camisa da pessoa), e responde a gestos do braço. As imagens capturadas do braço da pessoa são interpretadas por dois algoritmos distintos: template matching, que busca correlacionar a imagem capturada com uma série de imagens pré- armazenadas, e redes neurais, que retorna uma previsão dos ângulos do braço da pessoa com base na imagem capturada. Ambos os algoritmos são sucedidos pelo algoritmo de Viterbi (Viterbi (1967)), que relaciona a série de capturas de imagens estáticas a um gesto dinâmico pré-definido pelos pesquisadores na base de dados.

Cheng and Kuniyoshi (2000) obtêm a imitação dos braços de um usuário por meio da segmentação de cor da pele. A estimação da posição da cabeça é utilizada para orientar o movimento do robô para que o usuário fique dentro do alcance da câmera do robô, e a imitação se dava pela imitação dos ângulos do ombro e cotovelo.

Riley, Ude, Wade, and Atkeson (2003) usam marcos coloridos passivos na parte superior do corpo do usuário e câmeras externas e no robô a fim de obter a configuração do usuário, para que o robô o imite.

Nakaoka, Nakazawa, Yokoi, Hirukawa, and Ikeuchi (2003) usam uma série de câmeras espalhadas em uma sala para captar 30 marcadores posicionados no corpo do usuário, para que o robô imite o usuário dançando.

Molina-Tanco, Bandera, Marfil, and Sandoval (2005) também utilizam câmeras para detecção de cor de pele, e calculam os ângulos de articulações que são reproduzidos pelo

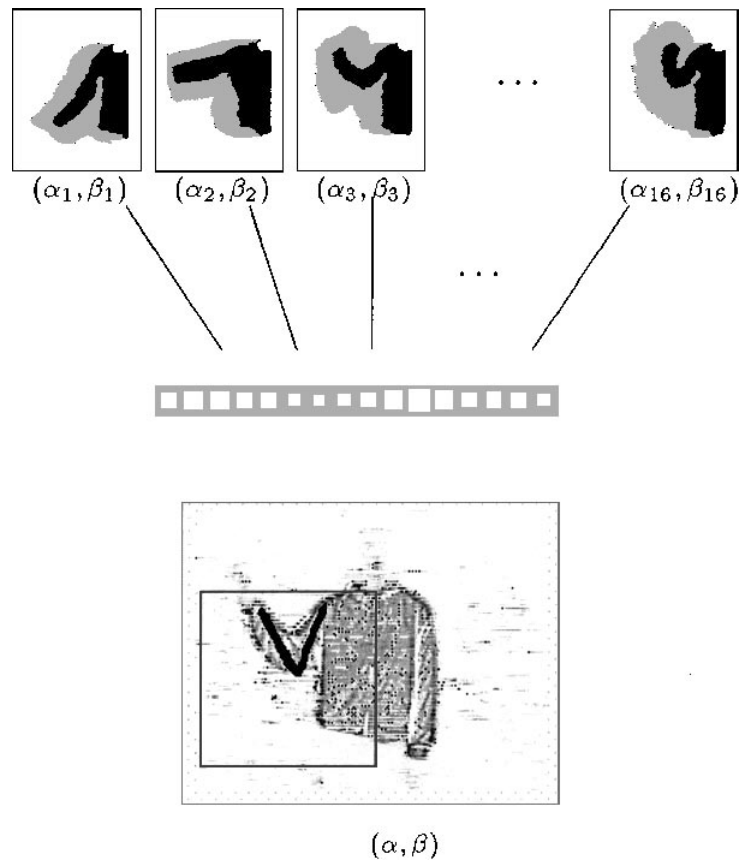


Figura 2.2: Rede neural identifica ângulos correspondentes à base de dados para reconhecimento de gestos, em Waldherr et al. (2000)

robô.

Ott, Lee, and Nakamura (2008) utilizam Modelos Ocultos de Markov para reconhecer gestos capturados por 34 marcadores no corpo do usuário, permitindo imitação de diversos movimentos pelo robô.

Suay and Chernova (2011) utilizam o Kinect para detectar movimentos dos braços do usuário, que assim controlava o olhar e os passos do robô.

Koenemann and Bennewitz (2012) usam uma série de sensores inerciais no corpo do usuário para passar para o robô o movimento das articulações, permitindo imitação em tempo real.

Zuher and Romero (2012) utilizam o robô NAO e o sensor Kinect para desenvolver uma interface para interação com seres humanos através de uma série de gestos pré-programados, e da imitação de movimentos do ser humano. Este trabalho compara os ângulos de algumas partes do corpo fornecida pela leitura do *Kinect Skeleton*, e então utiliza o framework OpenNI para repassar instruções ao NAO. Este processo é explicado em mais detalhes na seção de Metodologia.

2.2.1 Algoritmo de Viterbi

Proposto em 1967 para decodificar sinais digitais com ruído (Viterbi (1967)), o algoritmo de Viterbi é muito usado atualmente em telecomunicações, reconhecimento de

fala, interpretação de texto e bioinformática (sequenciamento de DNA). Ele é baseado em Modelos Ocultos de Markov, nos quais o estado do sistema é desconhecido, e pode somente ser inferido probabilisticamente com base em observações relacionadas.

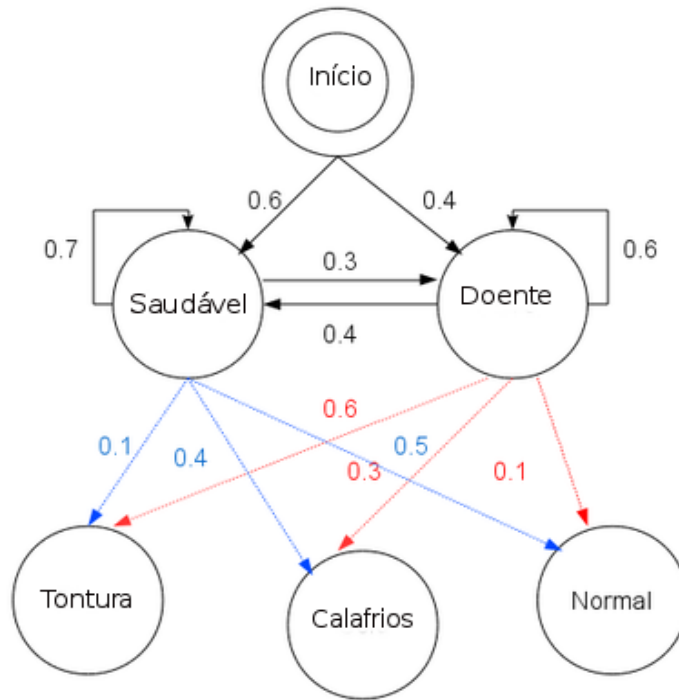


Figura 2.3: Modelo oculto de Markov

Fonte: Wikipedia

Os estados ocultos seguem uma cadeia de Markov, na qual o há uma probabilidade de transição de um estado no instante t para outro estado no instante $t+1$. Por exemplo, um paciente pode estar doente ou saudável (estes são os dois estados possíveis), mas o médico só observa alguns sintomas (tontura, calafrio, normal), e com base neles deve determinar o estado do paciente em cada dia.

Há uma probabilidade de o paciente passar de saudável para doente e vice-versa com o passar do tempo, e há uma probabilidade de cada sintoma dependendo do estado do paciente naquele dia. O algoritmo de Viterbi é um algoritmo de programação dinâmica recursivo que, dada uma sequência de observações e as probabilidades associadas, determina a sequência de estados com a maior probabilidade de gerar a sequência observada. Para isso, o algoritmo calcula recursivamente, para cada instante t e para cada estado possível k em t , a sequência de estados anteriores a t com a maior probabilidade de terminar naquele estado. Sejam π_i a probabilidade inicial de cada estado i , $a_{i,j}$ a probabilidade de transicionar do estado i para o estado j , e y_1, y_2, \dots, y_T as observações em cada instante t . Então, a sequência de estados mais provável x_1, x_2, \dots, x_T é dada pela relação recursiva:

$$V_{1,k} = P(y_1|k) \times \pi_k$$

$$V_{t,k} = P(y_t|k) \times \max_x (a_{x,k} \times V_{t-1,x})$$

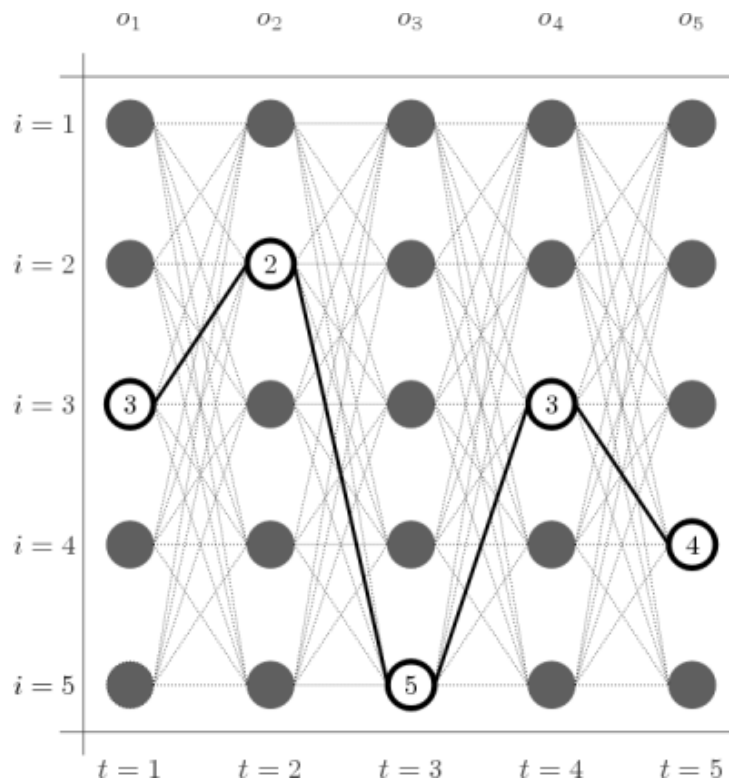


Figura 2.4: Algoritmo de Viterbi

Fonte: Wikipedia

Onde $V_{t,k}$ é a probabilidade da sequência de estados mais provável para as primeiras t observações que termina no estado k . Os estados de maior probabilidade então são dados por:

$$x_T = \operatorname{argmax}_x (V_{T,x})$$

$$x_{t-1} = \operatorname{Ptr}(x_t, t)$$

Onde $\operatorname{Ptr}(k, t)$ é um ponteiro que retorna o valor de x usado para computar cada $V_{t,k}$. A complexidade do algoritmo é cúbica (quadrada no número de estados, multiplicado pelo número de instantes t).

2.2.2 Algoritmo de Baum-Welch

O algoritmo de Baum-Welch é um algoritmo de maximização de expectativas usado para encontrar os parâmetros desconhecidos de um modelo oculto de Markov (Baum, Petrie, Soules, and Weiss (1970)). Este algoritmo é muito utilizado para processamento de fala, criptoanálise, modelagem probabilística de genoma, e reconhecimento de gestos.

Sejam X_t uma variável aleatória oculta com N possíveis estados, $A = \{a_{ij}\} = P(X_t = j | X_{t-1} = i)$ a matriz de transição estocástica, $\pi_i = P(X_1 = i)$ a distribuição inicial de estados, Y_t a variável observada com K possíveis valores, $Y = (Y_1 = y_1, Y_2 = y_2, \dots, Y_t = y_t)$ a sequência de observações, e $B = \{b_{ij}\} = P(Y_t = i | X_t = j)$ a matriz $K \times N$ de probabilidade de observações para cada estado oculto.

O modelo oculto de Markov pode então ser descrito por $\theta = (A, B, \pi)$, e o algoritmo

de Baum-Welch computa os parâmetros do modelo que maximizam a probabilidade da sequência observada, ou seja, $\theta^* = \max_{\theta} P(Y|\theta)$.

O algoritmo é inicializado aleatoriamente com condições iniciais $\theta = (A, B, \pi)$. Realiza-se então um *forward pass* onde são calculados $\alpha_i(t) = P(Y_1 = y_1, Y_2 = y_2, \dots, Y_t = y_t, X_t = i|\theta)$, ou seja, a probabilidade de se observar uma determinada sequência no instante t e estar no estado i . A computação é recursiva:

$$\alpha_i(1) = \pi_i b_i(y_1)$$

$$\alpha_i(t+1) = b_j(y_{t+1}) \sum_{i=1}^N \alpha_i(t) a_{ij}$$

No *backward pass*, calcula-se $\beta_i(t) = P(Y_{t+1} = y_{t+1}, \dots, Y_T = y_T | X_t = i, \theta)$, ou seja, a probabilidade da sequência restante no instante t e estado i . A computação também é recursiva:

$$\beta_i(T) = 1$$

$$\beta_i(t) = \sum_{j=1}^N \beta_j(t+1) a_{ij} b_j(y_{t+1})$$

Calcula-se então as probabilidades $\gamma_i(t)$ (probabilidade do estado i no instante t) e $\xi_{ij}(t)$ (probabilidade de estados i e j nos instantes t e $t+1$, respectivamente).

$$\gamma_i(t) = P(X_t = i | Y, \theta) = \frac{\alpha_i(t) \beta_i(t)}{\sum_{j=1}^N \alpha_j(t) \beta_j(t)}$$

$$\xi_{ij}(t) = P(X_t = i, X_{t+1} = j | Y, \theta) = \frac{\alpha_i(t) a_{ij} \beta_j(t+1) b_j(y_{t+1})}{\sum_{k=1}^N \sum_{l=1}^N \alpha_k(t) a_{kl} \beta_l(t+1) b_l(y_{t+1})}$$

Os parâmetros de θ então são atualizados:

$$\pi_i^* = \gamma_i(1)$$

$$a_{ij}^* = \frac{\sum_{t=1}^{T-1} \xi_{ij}(t)}{\sum_{t=1}^{T-1} \gamma_i(t)}$$

$$b_i^*(k) = \frac{\sum_{t=1}^T 1_{y_t=v_k} \gamma_i(t)}{\sum_{t=1}^T \gamma_i(t)}$$

Estes passos são repetidos iterativamente até convergirem com uma tolerância adequada. A convergência ocorre em um máximo local.

2.2.3 Máquinas de estados finitos

Modelos ocultos de Markov têm sido muito usados para reconhecimento visual de gestos, sendo treinados com sequências de poses alinhadas temporalmente, e usando programação dinâmica para reconhecer uma nova sequência observada (Yamato et al. (1992)).

Máquinas de estados finitos são uma alternativa mais recente aos modelos ocultos de Markov, e apresentam a vantagem de não precisar de um conjunto muito grande de dados para treinamento (Hong, Turk, and Huang (2000a)). Para o sistema aqui proposto, a grande vantagem de uma máquina de estados finitos é permitir a correção de um movimento parcial incorreto executado pelo paciente, até que a transição para o próximo estado seja feita corretamente.

Bobick and Wilson (1997) modelam um gesto como uma sequência de estados em um espaço de configurações. Os dados de treinamento são manualmente segmentados e alinhados temporalmente. O reconhecimento de gestos é feito usando *fuzzy logic* e programação dinâmica para computar as poses médias que fazem parte de um gesto.

Davis and Shah (1994) modelam quatro fases qualitativamente distintas para um gesto manual qualquer: (i) posição inicial estática; (ii) movimento contínuo da mão e dedos durante o gesto; (iii) posição final estática; e (iv) movimento contínuo da mão de volta à posição inicial. O gesto é representado como uma lista de vetores e então comparado com os gestos-modelo já computados, baseado-se na distância dos vetores.

McKennah and Gong (1998) usam uma máquina de estados finita probabilística para reconhecer gestos. Os gestos são modelados como uma sequência de eventos visuais, onde cada evento é representado por um modelo probabilístico de trajetórias pré-segmentadas. A função de densidade probabilística para a duração de cada evento é estimada dos exemplos de treinamento como uma distribuição Gaussiana ou uniforme no intervalo relevante. As transições entre estados dependem tanto da verossimilhança do modelo observado quanto da função de densidade probabilística da duração do estado.

As técnicas mencionadas acima requerem segmentação e alinhamento manual dos dados para criar os estados que farão parte do modelo. Hong, Turk, and Huang (2000b) propõe uma técnica que auxilia na segmentação e alinhamento dos dados, sem envolver tanto trabalho manual humano. Eles modelam cada estado, em espaço espacial-temporal, como uma Gaussiana multivariada. O gesto como um todo é representado por uma máquina de estados finitos. Um *threshold* é pré-definido para permitir alguma variância espacial em cada estado. O número de estados é então calculado usando *dynamic k-means clustering* nos dados de treinamento, sem levar em conta a informação temporal. A informação temporal então é acrescentada aos estados, concluindo a segmentação e ordenamento temporal dos estados. Ao observar um novo movimento, a máquina de estados finitos decide se passa ao próximo estado baseado nos parâmetros espaciais observados e no tempo da observação. Se a máquina de estados finitos atingir o estado final, aquele gesto é dito reconhecido.

2.3 Teleoperação

O controle à distância de robôs chama-se teleoperação. Na teleoperação direta, o operador controla diretamente e em tempo real os atuadores do robô. Na teleoperação

coordenada, o operador também controla os atuadores do robô, mas seus comandos estão sujeitos à supervisão de um algoritmo interno do robô para por exemplo impedir que erros humanos causem acidentes. Na teleoperação com controle supervísório, os algoritmos do robô tomam as decisões para o controle de seus atuadores, e o operador humano emite instruções de mais alto nível. Um exemplo de teleoperação com controle supervísório é o uso de gestos pré-determinados para que o robô caminhe na direção desejada pelo operador, como em Zuher and Romero (2012).

A teleoperação vem sendo utilizada com sucesso em áreas como manipulação de materiais radioativos, exploração marítima, exploração espacial, robôs aéreos e terrestres de combate, vigilância, e cirurgia (Funda and Paul (1991), Madni, Chu, and Freedy (1983), Yoerger and Slotine (1987), Skaar and Ruoff (1994), Bejczy and Szakaly (1987), Imaida, Yokokohji, Oda, Yoshikawa, et al. (2004), McNickle (2012)).

A teleoperação deve levar em conta a correspondência de sensores e atuadores entre o robô e o operador, utilizando sensores como vestimentas mecânicas, leitores de movimentos dos olhos, mouses e joysticks. A comunicação entre operador e robô pode ser feita por meio de fios, wireless, rádio, infravermelho, desde que protocolos de comunicação adequados sejam estabelecidos. É importante também levar em conta o tempo de necessário desde a emissão de um comando pelo operador até a execução de ação correspondente pelo robô (*delay*) (Lichiardopol (2007)).

O sistema de teleoperação deve procurar manter a estabilidade, isto é, manter-se operacional apesar de condições adversas causadas pela conduta do operador ou por mudanças no ambiente. O sistema deve também prover um senso real de presença ao operador no ambiente remoto, chamado de telepresença (Hokayem and Spong (2006)).

Na década de 60, o comportamento *move-and-wait* era observado nos experimentos de Sheridan and Ferrell (1963) e Ferrell (1965) quando um sistema teleoperado era submetido a atrasos no loop de comunicação do sistema. Esses experimentos objetivavam a mensuração do tempo requerido para a realização de determinadas tarefas sob o viés dos atrasos. Esses ensaios revelaram a direta relação linear entre os atrasos induzidos no circuito e o tempo de execução das tarefas submetidas aos operadores, e uma satisfatória execução de movimentos realizados pelos operadores ante a estratégia *move-and-wait*.

Para superar o tempo de execução de teleoperação proporcional ao atraso no loop de comunicação da estratégia anterior, Ferrell and Sheridan (1967) propuseram que os comandos executados pelos operadores fossem tratados de forma supervísória, caracterizando-os sob natureza análoga ou simbólica. Assim, o mecanismo teleoperado deveria ser desenvolvido com certa autonomia para execução das pequenas tarefas compreendidas pelos comandos supervísórios. Whitney (1969) sugeriu uma abordagem supervísória com enfoque na otimização das tarefas realizadas pelo sistema, podendo assim estimar o meio pelo qual se obteria desempenho ótimo na realização de tarefas durante a teleoperação. Madni et al. (1983) utilizaram uma linguagem de programação voltada ao desenvolvimento de controle supervísório com algumas características de telepresença, tais como visual e força.

Autonomia é um conceito importante para a teleoperação, que pode ser entendida como o número de solicitações de um robô ao operador durante a comunicação, ou pelo volume de tempo que um robô pode ser ignorado Goodrich and Schultz (2007). Segundo De Barros and Lindeman (2008), os níveis de autonomia de robôs para diferentes modos de operação na literatura podem ser classificados como completamente controlado, controle compartilhado (teleoperação segura, semiautônomo e alto nível de teleoperação) e completamente autônomo.

Neste trabalho, o controle será compartilhado, já que cabe ao robô decidir como reagir e se mover, dados os movimentos do usuário. A teleoperação não será remota, já que é importante que o paciente aprenda os movimentos observando o robô. O robô não simplesmente imitará os movimentos do paciente, mas os observará e corrigirá caso estejam incorretos. De qualquer forma, os conceitos de teleoperação discutidos acima se aplicam para que o sistema funcione de forma fluida e sem frustrar o usuário.

Materiais e metodologia

3.1 Robô NAO

Desenvolvido pela empresa francesa Aldebaran Robotics a partir de 2004, e disponibilizado para institutos educacionais e de pesquisa no final de 2008, o robô NAO atualmente é utilizado em mais de 350 universidades e laboratórios de pesquisa em todo o mundo, e é vendido (somente para fins de pesquisa e educação) por US\$16.000, não incluídos aí os impostos de importação brasileiros, que dobram este custo.

Do tamanho de uma criança pequena (58 cm), o robô possui 25 graus de liberdade em suas articulações, permitindo uma gama de movimentos semelhante (mas com algumas restrições) a de um ser humano.

Estes movimentos podem ser controlados através do kit de desenvolvimento de software (SDK), compatível com 8 linguagens de programação, entre elas C++, Python e Java. A versão mais atual (2011) é controlada por uma CPU Intel ATOM de 1.6 GHz localizada na cabeça, rodando um kernel Linux e o middleware NAOqi.

Para comunicação, o NAO possui um sintetizador de voz, luzes LED nos olhos, peito e pés, e dois speakers de alta fidelidade. O NAO também possui vários sensores, entre eles duas câmeras, quatro microfones, sonar, e sensores infravermelho, inercial, tátil e de pressão.

A bateria do NAO tem duração aproximada de uma hora e meia. Embora o NAO possua sensores, eles não são tão apurados quanto os do sensor Kinect, e como para este projeto é fundamental a detecção precisa dos movimentos da criança, não utilizaremos os sensores do NAO.

Tanto o NAO quanto o Kinect se comunicarão com um computador, que interpretará os dados vindos do Kinect e transmitirá instruções ao NAO. A comunicação com o NAO

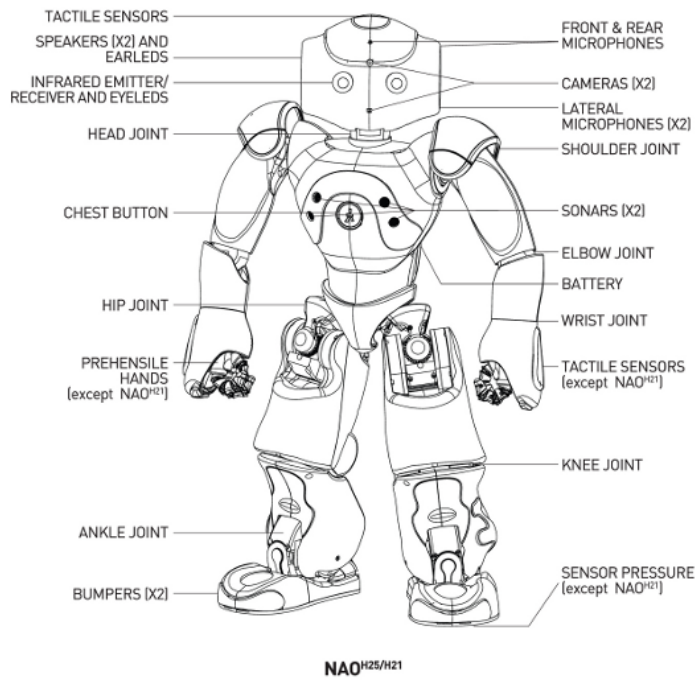


Figura 3.1: Componentes do robô NAO

Fonte: Aldebaran Robotics

pode ser feita por um cabo de Ethernet ou por WIFI (sem fio), e o Kinect exige um cabo USB.

O robô NAO possui uma interface de programação de aplicações (API) que permite controlá-lo facilmente, uma vez instalado seu kit de desenvolvimento de software (SDK), com auxílio do software qiBuild. O SDK está disponível para C++ e Python. Neste projeto, foi utilizado o SDK em C++. Um exemplo simples de como fazer o robô falar é mostrado na figura 3.2. Para mover o robô, é necessário definir alguns parâmetros, como a rigidez dos motores nas juntas (*stiffness*), o nome da junta desejada, e a velocidade do movimento, como mostrado na figura 3.3.

```
#include <alproxies/altexttospeechproxy.h>

const std::string phraseToSay = "Hello_world";
AL::ALTextToSpeechProxy tts("nao.local" , 9559);
tts.say("Hello_world");
```

Figura 3.2: API para fala do robô NAO

Fonte: Aldebaran Robotics

```

#include <iostream>
#include <alerror/alerror.h>
#include <alproxies/almotionproxy.h>

int main(int argc, char* argv[]) {

    /** The name of the joint to be moved. */
    const AL::ALValue jointName = "HeadYaw";

    AL::ALMotionProxy motion(argv[1], 9559);

    /** Target stiffness. */
    AL::ALValue stiffness = 1.0f;
    /** Time (in seconds) to reach the target. */
    AL::ALValue time = 1.0f;
    /** Call the stiffness interpolation method. */
    motion.stiffnessInterpolation(jointName, stiffness, time);

    /** Set the target angle list, in radians. */
    AL::ALValue targetAngles = AL::ALValue::array(-1.5f, 1.5f, 0.0f);
    /** Set the corresponding time lists, in seconds. */
    AL::ALValue targetTimes = AL::ALValue::array(3.0f, 6.0f, 9.0f);
    /** Specify that the desired angles are absolute. */
    bool isAbsolute = true;

    /** Call the angle interpolation method. The joint will reach the
    * desired angles at the desired times.
    */
    motion.angleInterpolation(jointName, targetAngles, targetTimes, isAbsolute);

    /** Remove the stiffness on the head. */
    stiffness = 0.0f;
    time = 1.0f;
    motion.stiffnessInterpolation(jointName, stiffness, time);
}

```

Figura 3.3: API para movimentação do robô NAO

Fonte: Aldebaran Robotics

3.2 Sensor Kinect

Para uma melhor adaptação do paciente, é importante que os sensores utilizados sejam não-intrusivos, isto é, não necessitem que nenhum objeto seja colocado sobre o corpo do paciente, e obtenham informações sem tocá-lo. Um sensor não-intrusivo com ótimo custo benefício é o Kinect.

Lançado no final de 2010 pela Microsoft para controle do console de videogames Xbox via gestos e fala, o Kinect entrou para o livro de recordes da Guinness como o dispositivo eletrônico de consumo com venda mais rápida, vendendo 8 milhões de unidades nos primeiros 60 dias, e seu uso não ficou restrito a videogames.

Devido a sua ótima relação custo-benefício, o Kinect foi amplamente adotado para diferentes usos de hobby e pesquisa. Custando em torno de US\$150, a versão atual para Windows (2012) contém uma câmera colorida com resolução de 1280x960 pixels, sensor de profundidade infravermelho com distância mínima de 1.2 metro e máxima de 3.5 metros (é importante notar que o infravermelho emitido pelo sensor tem frequência próxima a da emitida pelo sol, e portanto ele não funciona em ambientes abertos durante o dia), quatro microfones direcionais, e um acelerômetro.

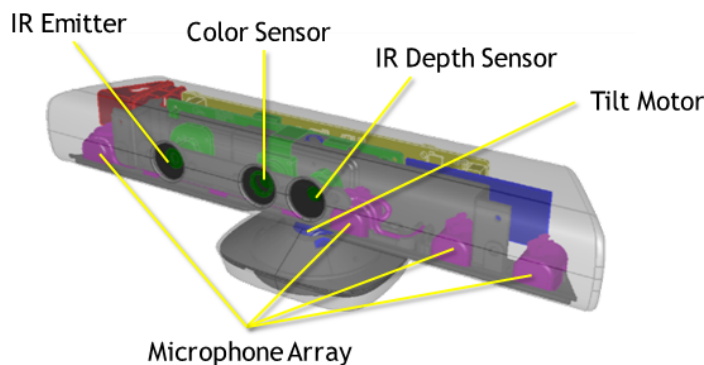


Figura 3.4: Componentes do Kinect

Fonte: Microsoft

Pesquisadores em diversas universidades e empresas vêm utilizando o Kinect para mapeamento robótico e interação humano-robô, para sistemas de vigilância que funcionam em escuridão total, para modelar o corpo do usuário para medir sua gordura corporal ou permitir que sua imagem fosse sobreposta com a imagem de diferentes peças de roupa, e para que o usuário possa tocar piano batendo os dedos em uma mesa vazia.

O Kinect também vem sendo usado para pesquisa em medicina, permitindo medidas objetivas para detectar autismo, desordem de déficit de atenção e transtorno obsessivo-compulsivo em crianças, e por cirurgiões para acessar imagens de tomografia e ressonância magnética sem ter que tocar um computador.

Embora os *drivers* do Kinect sejam proprietários e secretos, várias alternativas *open source* existem. Para este projeto, utilizaremos o *framework* OpenNI, que permite ler os dados dos sensores transmitidos pelo Kinect por seu cabo USB, e pode ser utilizado

tanto em sistemas operacionais Windows quanto em Linux. O *framework* OpenNI permite que um modelo 3D simplificado do esqueleto do usuário seja capturado pelos sensores do Kinect (*Kinect Skeleton*), retornando os ângulos e posições das principais articulações do corpo humano.

Livingston et al. [Livingston, Sebastian, Ai, and Decker (2012)] analisaram a performance do *Kinect Skeleton* em uma série de critérios: alcance, ruído, acurácia, latência e resolução. Os resultados obtidos foram satisfatórios para reconhecimento de gestos em partes maiores do corpo como braços e mãos, embora a acurácia não seja suficiente para reconhecer gestos de partes menores como os dedos. A acurácia e a latência medidas no Kinect foram piores que as de sistemas comerciais mais caros, mas os autores concluem que dado o custo muito menor do Kinect, o uso do *Kinect Skeleton* para reconhecimento de gestos é satisfatório.

3.3 Imitação de gestos

Para a imitação dos movimentos observados, o sistema usa a técnica desenvolvida em Zuher and Romero (2012), que utiliza a biblioteca OpenNI para capturar junções do corpo humano via sensor Kinect, e as mapeia em uma ação apropriada a ser realizada pelo robô NAO.

3.3.1 Fundamentos

Para calcular a configuração angular das junções nos movimentos de membros do corpo humano e reproduzi-las no robô, deve-se considerar três espaços de coordenadas:

Mundo: Espaço de coordenadas onde se estabelecem posições absolutas para objetos de interesse, servindo como referência para outros espaços de coordenadas.

Objeto: Espaço de coordenadas relativo ao objeto de interesse. Quando o objeto se move, o espaço de coordenadas também se move, tendo sua origem em um ponto pré-determinado no objeto.

Câmera: Espaço de coordenadas de objeto com origem na câmera (ou outro sensor, como o Kinect) usada para medir ângulos observados no mundo e em outros objetos. O sensor Kinect tem a vantagem, em relação a uma câmera, de já representar o objeto de interesse em um espaço tri-dimensional.

Para representar ângulos obtidos em um espaço de coordenadas em outro espaço de coordenadas, são utilizadas matrizes de rotação e ângulos de Euler:

Matriz de rotação: Usada para obter as coordenadas de um objeto em um novo espaço de coordenadas, a partir das coordenadas em outro espaço. As colunas de uma matriz de rotação podem ser interpretadas como vetores que formam a base do novo

espaço de coordenadas, em sua representação no espaço de coordenadas original. Como os espaços aqui tratados são tri-dimensionais, as matrizes de rotação são sempre 3x3.

Ângulo de Euler: São três ângulos propostos por Leonhard Euler para descrever a orientação de um corpo rígido, aqui chamados de *yaw* (rotação em torno do eixo vertical), *pitch* (rotação em torno do eixo transversal) e *roll* (rotação em torno do eixo longitudinal).

3.3.2 Mapeando articulações

Para obter-se a configuração angular em três dimensões de uma junção observada do corpo humano, cria-se a matriz de rotação para os três ângulos de Euler, conforme Dunn and Parberry (2011). A representação dos ângulos de Euler utilizada é do tipo canônico, com *yaw* e *roll* entre -180° e $+180^\circ$, e *pitch* entre -90° e $+90^\circ$.

As articulações humanas são classificadas de acordo com seus graus de liberdade para rotações. Articulações mono-axiais fazem apenas movimentos de flexão e extensão, como o joelho e o cotovelo. Articulações bi-axiais, além de flexão e extensão, também permitem movimentos de adução e abdução, como o punho. Articulações tri-axiais permitem, além dos movimentos já citados, o movimento de rotação, como no caso do ombro.

O robô NAO possui articulações semelhantes às do corpo humano, mas não com exatamente os mesmos graus de liberdade. O ombro do robô, por exemplo, não tem o grau de liberdade correspondente ao *yaw* do ombro humano, o que é compensado pelo *yaw* no cotovelo do robô, permitindo simular a rotação de todo o braço humano via a rotação de apenas o antebraço do robô. Assim, o *yaw* observado no ser humano é mapeado para o *yaw* do cotovelo do robô ao realizar uma imitação. O quadril do robô não permite a rotação independente das pernas, e o pulso do robô é fixo, com zero graus de liberdade, e portanto esse tipo de movimento simplesmente é ignorado pelo robô.

Os ângulos executados pelo robô para imitar cada junção observada são explicados abaixo, seguindo a notação de Zuher and Romero (2012):

- "*nome_da_junção_{eixo}*" corresponde à distância em milímetros da junção à origem do sistema de coordenadas da câmera, ao longo do eixo x, y ou z, medida pelo sensor Kinect. Por exemplo, *ombro_z* é a distância do ombro da origem em milímetros ao longo do eixo z, conforme observado pelo sensor Kinect. Caso o eixo não seja explicitado, a variável se refere somente ao comprimento em milímetros de um membro, como por exemplo *braço_superior*.
- "*NAO_junção_{ângulo}*" denota o valor de um dos três ângulos de Euler (*yaw*, *pitch* ou *roll*) de uma junção do robô NAO, como por exemplo *NAO_ombro_{pitch}*. Acrescentando-se "*_limite*" à variável para se referir ao ângulo máximo possível para aquela junção.

Seguindo a notação acima, os ângulos de Euler de interesse então são obtidos a partir da matriz de rotação, exceto em alguns casos onde este cálculo se revelou instável durante o movimento devido a limitações do sensor Kinect. Nestes casos, o uso das regras de três e inversas do cosseno descritas abaixo se mostraram uma alternativa mais adequada para o cálculo dos ângulos de Euler.

Pescoço O *yaw* e o *pitch* do pescoço são estimados pela transformação da matriz de rotação do pescoço em ângulos de Euler.

Ombro O *pitch* e o *roll* do ombro são calculados seguindo as regras de três dadas abaixo.

$$NAO_ombro_{pitch} = \frac{(ombro_z - cotovelo_z) \times NAO_ombro_{pitch_limite}}{braço_superior}$$

$$NAO_ombro_{roll} = \frac{(ombro_x - cotovelo_x) \times NAO_ombro_{roll_limite}}{braço_superior}$$

Cotovelo O *yaw* do cotovelo é calculado pela transformação da matriz de rotação do cotovelo em ângulos de Euler. O *roll* do cotovelo é calculado como a inversa do cosseno do ângulo formado pelas junções ombro, cotovelo e mão.

$$NAO_cotovelo_{roll} = \cos^{-1} \left(\frac{braço_superior^2 + antebraço^2 - segmento_ombro_mão^2}{2 \times braço_superior \times antebraço} \right)$$

Quadril O *pitch* do quadril é calculado pela transformação da matriz de rotação do quadril em ângulos de Euler. O *roll* do quadril é calculado via regra de três abaixo.

$$NAO_quadril_{roll} = \frac{(quadril_x - joelho_x) \times NAO_quadril_{roll_limite}}{perna_superior/2}$$

Joelho O *pitch* do joelho é calculado como a inversa do cosseno do ângulo formado pelas junções quadril, joelho e pé.

$$NAO_joelho_{pitch} = \cos^{-1} \left(\frac{perna_superior^2 + perna_inferior^2 - segmento_quadril_pé^2}{2 \times perna_superior \times perna_inferior} \right)$$

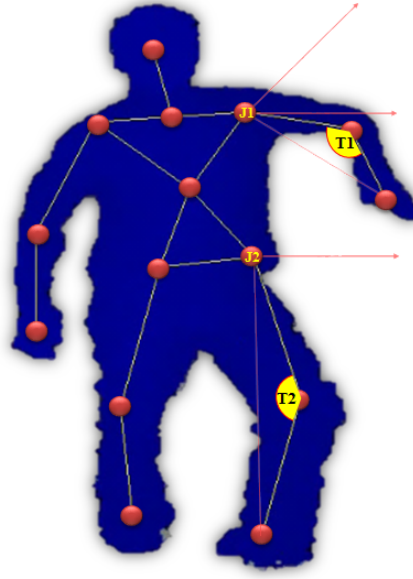


Figura 3.5: Junções e ângulos considerados na imagem

Fonte: Zuher and Romero (2012)

3.4 Algoritmo *k-means*

K-means clustering particiona as poses capturadas em k grupos (*clusters*), onde cada observação pertence ao grupo com posição média mais próxima. Embora este seja um problema NP-difícil, uma heurística eficiente permite encontrar uma solução que converge rapidamente para um ótimo local. Dadas n poses observadas $p = \{p_1, p_2, \dots, p_n\}$, onde cada pose é um vetor com quatro dimensões (espacial-temporal), o objetivo do algoritmo é encontrar uma partição destas n observações em k ($k \leq n$) grupos $S = \{S_1, S_2, \dots, S_k\}$ de forma a minimizar a soma dos quadrados intra-grupo (onde o desvio do centroide é dado pela distância Euclidiana):

$$\arg \min_S \sum_{i=1}^k \sum_{p \in S_i} \|p - \mu_i\|^2$$

onde μ_i é média dos vetores em S_i .

Para tal, foi utilizado o algoritmo de Lloyd (MacKay (2003)), que seleciona aleatoriamente k centroides $m_1^{(1)}, \dots, m_k^{(1)}$ e então repete iterativamente os seguintes passos:

Agrupamento Designe cada pose ao grupo do centroide mais próximo, de forma a minimizar a soma dos quadrados intra-grupo.

$$S_i^{(t)} = \left\{ p_p : \left\| p_p - m_i^{(t)} \right\|^2 \leq \left\| p_p - m_j^{(t)} \right\|^2, \forall j, 1 \leq j \leq k \right\}$$

onde cada pose p_p é designada a somente um grupo, com empates sendo decididos arbitrariamente.

Atualização Calcule a média dos vetores de cada grupo encontrado no passo anterior, e use o valor encontrado como o novo centroide daquele grupo.

$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{p_j \in S_i^{(t)}} p_j$$

Este novo centroide minimiza a soma dos quadrados do seu grupo, já que a média é o estimador de mínimos quadrados.

O processo termina quando os centroides convergem para um ótimo local e não mudam mais com novas iterações. Como ambos os passos acima minimizam a soma dos quadrados intra-grupo, e como existe um número finito de partições possíveis, o algoritmo sempre converge para uma solução ótima local, mas não há garantia que esta solução seja um ótimo global.

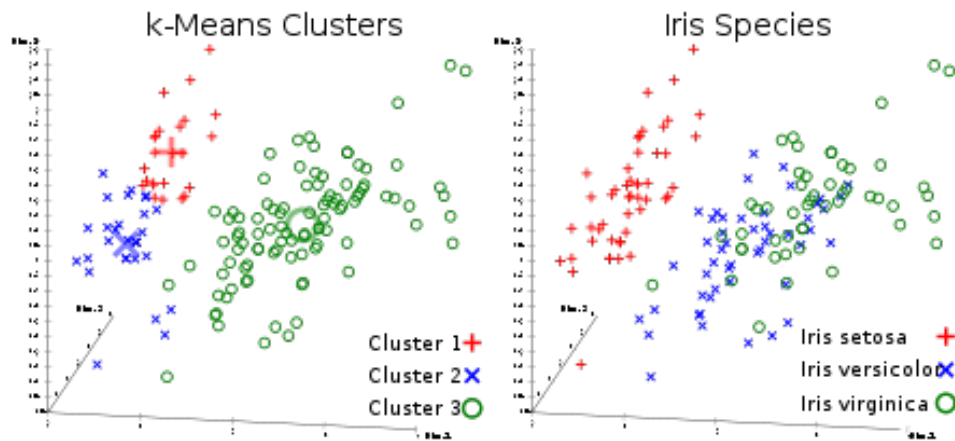


Figura 3.6: Algoritmo k-means identificando agrupamentos distintos

Fonte: Wikipedia

3.5 Reconhecimento de emoções via modelagem geométrica de faces

Para avaliar o estado emocional do paciente durante o tratamento, neste trabalho tem sido utilizado o sistema desenvolvido em Libralon and Romero (2013) para reconhecimento de emoções via modelagem geométrica de rostos. Foram capturadas múltiplas imagens do rosto de cada paciente durante o experimento e as emoções detectadas nas imagens foram usadas para avaliar o estado emocional do paciente durante o tratamento.

As expressões faciais são as mudanças no rosto de uma pessoa como resposta a seu estado emocional interno, intenções, ou comunicação social e são uma área ativa de pesquisa comportamental. Ela requer medidas de movimento facial e reconhecimento de expressões, podendo ser analisada via posição estática do rosto ou via movimento de músculos da face. Uma teoria muito aceita é que há um conjunto básico de emoções que foram

preservadas durante o processo evolucionário por permitir a adaptação do comportamento do organismo a situações específicas do dia-a-dia.

Libralon and Romero (2013) discutiram o reconhecimento de emoções com base em elementos faciais, propondo diferentes conjuntos de características para representar diferentes elementos da face humana e os avaliaram usando técnicas de aprendizado de máquina. Seus resultados indicam que este é um método válido para identificação de emoções.

Os métodos baseados em características geométricas (métodos computacionais) são utilizados na modelagem facial devido à existência de evidências psicológicas indicando que esta abordagem é mais semelhante à maneira como os seres humanos interpretam os elementos da face.

Desse modo, seis diferentes representações faciais são propostas para a identificação das emoções. Estas representações são capazes de codificar a configuração facial apresentada durante a expressão de uma emoção, assim como as diferenças entre esta configuração em relação a um modelo de face neutra, que não apresenta emoção.

O sistema de visão utilizado para a obtenção das informações faciais é um aperfeiçoamento do sistema chamado de Face Tracker. Os elementos considerados pelas representações propostas mapeiam as regiões faciais que estão essencialmente relacionadas a movimentos realizados quando da expressão de uma emoção, de acordo com estudos psicológicos (Ekman and Friesen (2003)).

Para obtê-los, o Face Tracker foi modificado para mapear apenas um subconjunto dos 66 pontos faciais inicialmente obtidos, composto por 33 pontos, que estão distribuídos da seguinte maneira: oito mapeiam os lábios (boca), doze pontos mapeiam os olhos, seis para cada um deles. As narinas são representadas por dois pontos, assim como três pontos delimitam cada uma das sobrancelhas. O contorno facial é delimitado por cinco pontos característicos, sendo três para mapear o queixo, um para a extremidade direita, e o último para a extremidade esquerda da face. Para representar o formato dos olhos, da boca, e de regiões faciais que ativam movimentos musculares emocionais, existem oito áreas. Além disso, diferentes cálculos de ângulos e distâncias entre os 33 pontos mapeados completam as representações propostas.

Na Figura 3.7 podem ser visualizados os 33 pontos considerados na análise de emoções da face propostos em Libralon and Romero (2013), Libralon and Romero (2014).

A identificação de características faciais (métodos psicológicos) utiliza nove diferentes algoritmos de seleção, que são aplicados às seis representações faciais, para obter as características de diagnóstico de cada uma das emoções investigadas, além da expressão neutra. Os algoritmos de seleção investigados, selecionados para garantir generalização aos resultados obtidos, são algoritmo genético, *forward selection*, *backward elimination*, *linear forward selection*, algoritmo por enxame de partículas, busca scatter search, e ranqueamento de atributos utilizando os algoritmos *relief*, chi-quadrado e SVM.

Para garantir que as características faciais de diagnóstico escolhidas pelos algorit-

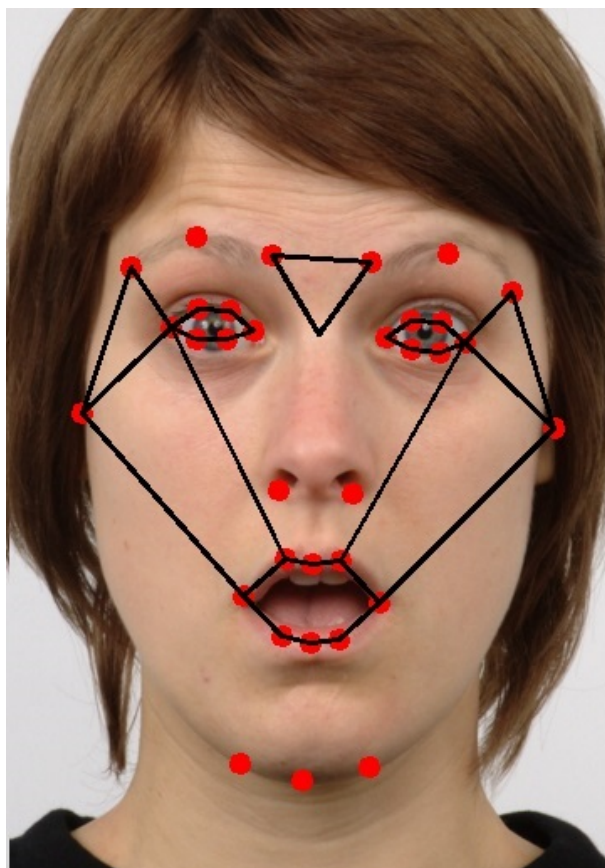


Figura 3.7: Pontos considerados para análise de emoções

Fonte:Libralon and Romero (2014)

mos de seleção estejam relacionadas apenas ao processo de reconhecimento de uma única emoção, cada uma das emoções estudadas é individualmente analisada, ou seja, um procedimento de seleção de características é realizado para cada emoção. A diversidade de paradigmas de aprendizado dos algoritmos de seleção avaliados propicia a criação de diferentes subconjuntos de características.

A análise das seis representações faciais foi feita pelos classificadores induzidos pelas SVMs, algoritmo C4.5 e redes MLP, obtidos a partir dos experimentos previamente realizados. Para a classificação final, os resultados individuais apresentados pelos diferentes classificadores foram combinados utilizando a estratégia de votação por maioria, obtendo uma única expressão emocional, dentre sete possibilidades, para cada representação facial.

Para as características de diagnóstico, os classificadores analisam as sete possíveis expressões emocionais investigadas, mas permitindo a escolha de quaisquer delas, inclusive a existência de empates. Para obter a predição final, os resultados encontrados pelos diferentes classificadores foram combinados pela estratégia de votação por maioria. Para ambos os métodos, computacional e psicológico, se a confiança obtida para a expressão emocional em análise é superior a um determinado limiar, este resultado é aceito, e ocorre a classificação. Se este limiar não é atingido ou se valores iguais de confiança são obtidos para diferentes expressões emocionais, uma lista das classificações encontradas é disponibilizada.

A modelagem obtém, em alguns casos, mais de uma classificação para as expressões emocionais investigadas. A não identificação da expressão em análise pode ocorrer devido ao mapeamento de uma expressão facial desconhecida ou a erros de reconhecimento. Uma vez que a modelagem não é capaz de determinar a razão da falha, a lista de possíveis classificações é disponibilizada para permitir, a um sistema ou arquitetura que a utilize, avaliar os resultados encontrados.

3.6 Considerações Finais

Neste capítulo foram apresentadas uma descrição do robô humanoide, NAO, bem como do sensor utilizado. Em seguida, foram descritos os métodos para imitação de gestos, agrupamento de dados e sobre modelagem geométrica da face para reconhecimento de emoções, que servirão de base para o método proposto no próximo capítulo.

Método Proposto

Neste Capítulo, inicialmente, um método para modelamento de gestos usando máquinas de estados finitos é proposto. Em seguida, um sistema integrado para captura, análise dos gestos e interação com o paciente, é apresentado. Finalmente, apresenta-se os vários experimentos realizados para testar o desempenho do sistema integrado proposto para auxílio de pacientes em sessões de fisioterapia e uma avaliação dos resultados é também apresentada.

4.1 Modelando gestos via máquinas de estados finitos

Um *gesto* é composto por uma série de *poses*, onde cada pose é uma imagem estática capturada ao longo do tempo enquanto o gesto é realizado.

Para que o sistema aprenda um movimento correto, um fisioterapeuta executa o movimento na frente do sensor Kinect. O sensor captura dez poses por segundo e as converte em uma mapa tri-dimensional da posição das principais articulações do corpo, usando o API do *Kinect Skeleton*.

As poses são então convertidas em uma sequência de pontos em espaço espacial-temporal e agrupadas via *k-means clustering* para transformar um único movimento complexo em uma série de movimentos parciais mais simples, que podem então ser incrementalmente demonstrados ao paciente e corrigidos mais rapidamente do que se somente o movimento completo fosse avaliado. O valor de k determina o número estados na máquina de estados finitos, e portanto o número de movimentos parciais nos quais o movimento completo será decomposto. O valor de k deve ser escolhido para gerar a melhor experiência para o paciente; se k for muito alto o paciente terá que pausar muitas vezes enquanto aguarda a avaliação e demonstração do próximo movimento parcial; se k for muito baixo

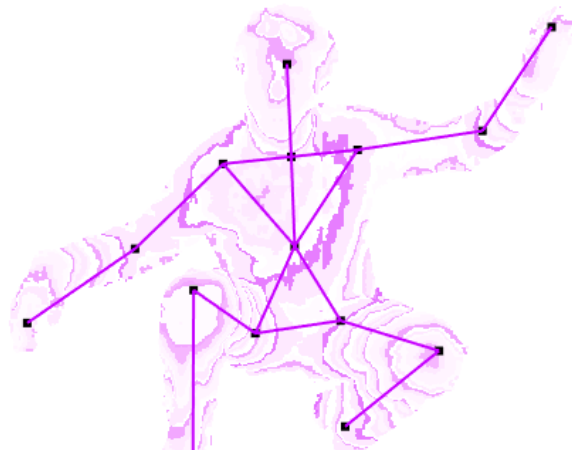


Figura 4.1: Kinect Skeleton

Fonte: Microsoft

o paciente terá que fazer movimentos muito amplos de uma vez e pode não entender onde cometeu eventuais erros.

4.1.1 Construção da Máquina de Estados Finitos

Adaptando a técnica de Hong et al. (2000b), cada estado s_i tem suas coordenadas espaço-temporais resumidas pelos parâmetros

$$\langle \bar{\mu}_i, \Sigma_i, d_i, T_i^{min}, T_i^{max} \rangle$$

onde

- $\bar{\mu}_i$ é um vetor espacial de centroides de cada junta do corpo naquele estado,
- Σ_i é a matriz de covariância espacial,
- d_i é um parâmetro de tolerância para decidir, em conjunto com a variância, se o estado foi atingido,
- e a duração do estado é dada por $[T_i^{min}, T_i^{max}]$.

Cada estado corresponde a um estado oculto em um modelo de Markov, cuja variância é dada pela dispersão das poses no grupo determinado pelo algoritmo de agrupamento *k-means* 3.4. A variância ao redor do centroide é usada para determinar se o estado foi atingido, em conjunto com o parâmetro d_i , que pode ser ajustado para melhorar a

performance do reconhecimento de gestos. O intervalo de tempo provê informação sobre a velocidade desejada e a ordem dos estados.

Obtidos os k centroides, constrói-se então o vetor espacial $\bar{\mu}_i$ em três dimensões que representa um estado da máquina de estados finitos, removendo-se a dimensão temporal do centroide. Calcula-se a matriz de variância 3×3 do grupo definido por cada centroide, para obter-se a matriz de covariância espacial Σ_i . Por fim, os tempos de duração mínimo e máximo de cada estado $[T_i^{min}, T_i^{max}]$ são calculados como frações do intervalo de tempo que contém todas as poses contidas naquele grupo. A tolerância pode ser calibrada experimentalmente para obter melhores resultados.

Com a máquina de estados finitos assim construída, o robô então demonstra cada movimento parcial ao paciente (respeitando-se restrições de mobilidade e equilíbrio do robô), executando em ordem as poses determinadas pelos centroides de cada estado (Zuher and Romero (2012)).

A transição de um estado para o próximo é o movimento que de fato o paciente deve executar. O robô executa cada transição de um estado para o próximo, pausa, e pede que o paciente o imite. Caso o sistema avalie que a nova pose do paciente atingiu o novo estado corretamente, o robô passa então ao próximo movimento.

Caso contrário, o robô informa ao paciente que o movimento foi realizado incorretamente, pede que o paciente tente novamente, e sugere uma correção pré-programada baseada na discrepância entre a pose observada e o estado desejado (e.g. "erga mais o braço esquerdo"). Ao atingir o estado final, o robô parabeniza o paciente, e pode sugerir que o paciente descanse ou pedir que o exercício seja repetido. Na Figura 4.2 pode ser visto uma criança imitando o gesto feito pelo robô durante uma sessão de fisioterapia.

Este uso de máquina de estados finitos ajuda a resolver o problema fundamental do reconhecimento de gestos: o mesmo movimento em geral não é realizado de forma idêntica por pessoas diferentes, ou mesmo pela mesma pessoa. Sempre existirão diferenças anatômicas, de posição, e de velocidade entre uma execução do gesto e outra. A variância computada em torno do vetor de centroides de articulações, que define cada estado provê a tolerância necessária para aceitar diferenças anatômicas e de posição para um dado gesto executado em diferentes condições, enquanto que a tolerância de tempo mínimo e máximo para a transição entre estados provê a tolerância para diferenças de velocidade de execução do gesto.

A alternativa mais comum para detecção de gestos seria avaliar o exercício como um todo por meio de um modelo oculto de Markov (HMM), para então informar ao paciente se o exercício foi executado corretamente ou não, mas isso tornaria difícil para o paciente saber onde ocorreu o erro na imitação do gesto.



Figura 4.2: Robô NAO guia paciente em sessão de fisioterapia

4.2 Sistema Integrado para Interação com o Paciente

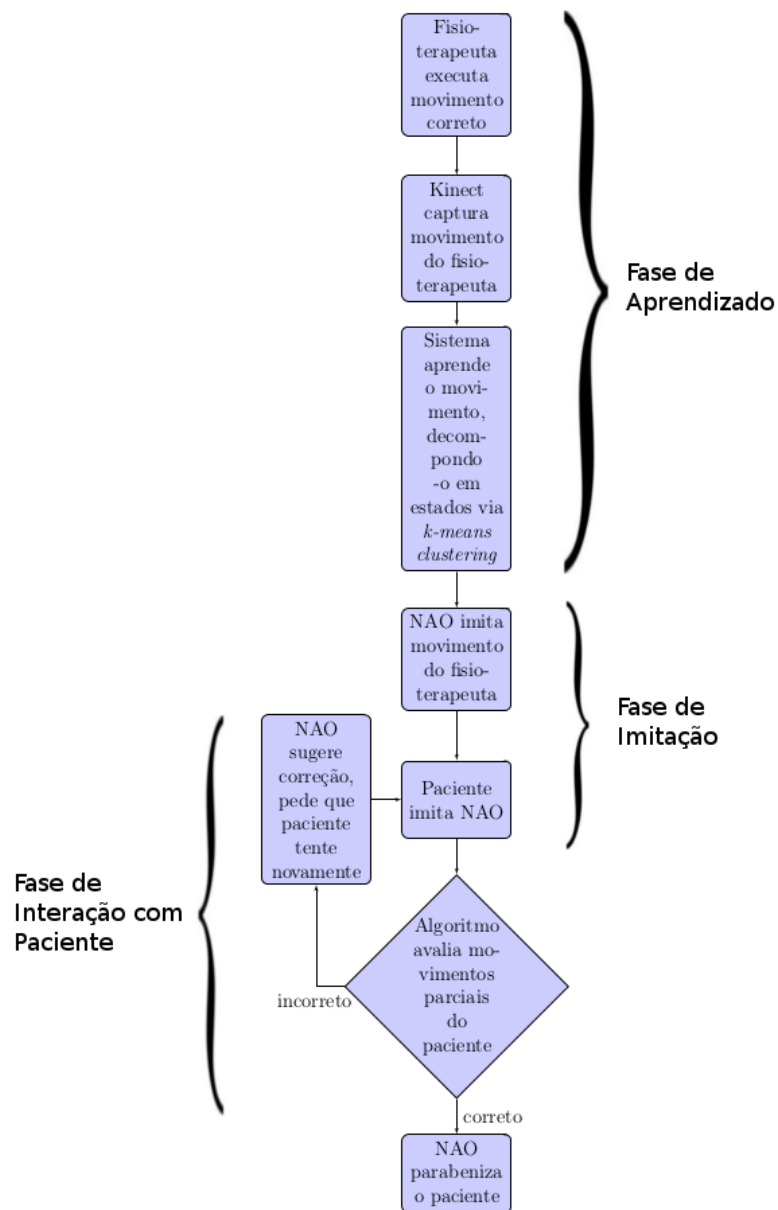


Figura 4.3: Interação entre fisioterapeuta, robô, e paciente para um exercício

O Sistema integrado para a interação entre o fisioterapeuta, o robô NAO e o paciente pode ser visualizado na Figura 4.3. Ele funciona da seguinte maneira:

- i. O fisioterapeuta executa o movimento completo e correto em frente ao sensor Kinect, que captura dez poses por segundo do movimento e as transforma em um vetor de posições de articulações do corpo.
- ii. As poses são agrupadas no espaço espacial-temporal via *k-means clustering*, de acordo com o número k apropriado de estados, e convertidas em uma máquina de estados finitos como explicado na seção anterior.

- iii. Com a máquina de estados finitos pronta, o robô NAO pode instruir pacientes a executar o movimento quantas vezes for desejado, executando cada transição entre estados e pedindo que o paciente o imite.
- iv. O sensor Kinect captura os movimentos do paciente, e o sistema decide se as poses observadas correspondem à transição apropriada entre estados da máquina de estados finitos.
- v. Em caso negativo, o robô sugere uma correção e pede que o paciente tente novamente.
- vi. Em caso positivo o robô demonstra a próxima transição entre estados e pede que o paciente o imite.
- vii. Quando todos os movimentos parciais são completados corretamente, o robô parabeniza o paciente.

4.3 Avaliação do Sistema Integrado

Para avaliar o desempenho do sistema robótico de fisioterapia, foram realizados experimentos com dez pacientes em duas clínicas de fisioterapia: a clínica do Departamento de Fisioterapia da Universidade Federal de São Carlos, e a clínica particular Fisioterapia São Carlos, com o auxílio de quatro fisioterapeutas destas clínicas.

Os pacientes testados foram cinco adultos e cinco crianças com idades entre seis e catorze anos. Os pacientes tinham dificuldades leves de movimento causadas por paralisia cerebral leve (seis pacientes), mal de Parkinson (dois pacientes) e imobilização para recuperação de fratura do braço (dois pacientes). Não foram considerados pacientes com problemas em outras regiões do corpo, apenas no braço.

O exercício consistia-se de uma série de movimentos de ambos os braços em torno das articulações dos ombros, de forma assimétrica (movimentos diferentes em cada braço), com o paciente em frente ao robô NAO imitando-o de forma espelhada. O robô ficava no chão ou em cima de uma mesa, dependendo do ambiente, com o Kinect ao seu lado capturando os movimentos do paciente. O exercício foi dividido em sete estados ($k = 7$), com cada estado correspondendo a um movimento de cerca de 90 graus da articulação do ombro, como por exemplo erguer o braço esquerdo lateralmente, baixar o braço direito frontalmente. O intervalo de tempo tolerado para a realização de cada movimento parcial $[T_i^{min}, T_i^{max}]$ utilizado foi entre 50% e 200% do tempo observado para cada movimento parcial realizado pelo fisioterapeuta durante a fase de aprendizado.

Uma vez completado o exercício inteiro corretamente, o robô pedia ao paciente que o fizesse outra vez e repetia o processo. Quando o paciente completava o exercício corretamente pela segunda vez, o robô parabenizava o paciente e dizia que ele podia descansar, encerrando a sessão.

Caso um movimento parcial fosse realizado incorretamente, o robô sugeria uma correção pré-programada baseada na discrepância entre o estado desejado e a pose observada do paciente (e.g., "abaixe mais o braço direito").

Os experimentos permitiram medir qualidade do algoritmo de avaliação de movimentos, comparando a avaliação do algoritmo com a avaliação do fisioterapeuta humano em cada movimento parcial, para classificar cada avaliação do algoritmo como Verdadeiro Positivo (T_P), Verdadeiro Negativo (T_N), Falso Positivo (F_P), e Falso Negativo (F_N).

Estas classificações foram então combinadas para medir a acurácia ($\frac{T_P+T_N}{Total}$), precisão ($\frac{T_P}{T_P+F_P}$), cobertura ou *recall* ($\frac{T_P}{T_P+F_N}$), e *F-measure* ($\frac{2}{\frac{1}{precisão} + \frac{1}{cobertura}}$), definidos em Makhoul, Kubala, Schwartz, Weischedel, et al. (1999).

Foi avaliado também o estado emocional dos pacientes antes, durante, e depois de cada experimento. Para avaliar e comparar o estado emocional do paciente antes e depois de cada experimento, foi utilizado um simples questionário pedindo que cada paciente dê uma nota de um a quatro ao seu estado emocional naquele momento, permitindo avaliar o impacto do experimento no estado emocional do paciente.

Para avaliar o estado emocional do paciente durante o tratamento, foi utilizado o sistema desenvolvido em Libralon and Romero (2013) para reconhecimento de emoções via modelagem geométrica de rostos. Foram capturadas múltiplas imagens do rosto de cada paciente durante o experimento, e as emoções detectadas nas imagens foram usadas para avaliar o estado emocional do paciente durante o tratamento.

4.3.1 Avaliação do algoritmo de reconhecimento de gestos

Dos dez pacientes, nove realizaram dois exercícios completos, sendo cada exercício composto pelos sete movimentos parciais discutidos anteriormente, mais as repetições de movimentos incorretos, resultando em um total de 192 avaliações. A avaliação de cada movimento parcial pelo algoritmo (como correto ou incorreto) foi comparada com a avaliação do mesmo movimento por um fisioterapeuta humano. Quando a avaliação do fisioterapeuta foi igual à do algoritmo, considerou-se que o algoritmo acertou; caso contrário, que o algoritmo errou. Mais especificamente, chamou-se de Verdadeiro Positivo (T_P) quando o algoritmo acertou a avaliação de um movimento correto, de Verdadeiro Negativo (T_N) quando o algoritmo acertou a avaliação de um movimento incorreto, de Falso Positivo (F_P) quando o algoritmo errou a avaliação de um movimento incorreto (ou seja, avaliou um movimento incorreto como correto), e de Falso Negativo (F_N) quando o algoritmo errou a avaliação de um movimento correto (ou seja, avaliou um movimento correto como incorreto).

Os resultados acima podem ser resumidos utilizando-se métricas para avaliação de algoritmos como acurácia (fração do total avaliada corretamente), precisão (de todos os movimentos avaliados como corretos, qual fração era de fato correta), e cobertura ou *recall* (de todos os movimentos corretos, qual fração foi avaliada como correta).

		Avaliação do Algoritmo		
		Correto'	Incorreto'	Total
Avaliação do Fisioterapeuta	Correto	T_P 112	F_N 21	133
	Incorreto	F_P 15	T_N 44	59
Total		127	65	192

Tabela 4.1: Matriz de confusão com resultados da avaliação do algoritmo

Com os resultados apresentados na tabela acima, obtemos:

$$acurácia = \frac{T_P + T_N}{Total} = \frac{112 + 44}{192} = 0.813$$

$$precisão = \frac{T_P}{T_P + F_P} = \frac{112}{112 + 15} = 0.882$$

$$cobertura = \frac{T_P}{T_P + F_N} = \frac{112}{112 + 21} = 0.842$$

$$F\text{-measure} = \frac{2}{\frac{1}{precisão} + \frac{1}{cobertura}} = \frac{2}{\frac{1}{0.882} + \frac{1}{0.842}} = 0.862$$

A acurácia pode não ser uma boa medida se, por exemplo, o número de resultados negativos for muito maior que o de positivos. Neste caso, um algoritmo que retorne sempre negativo terá alta acurácia, mas é inútil. Por isso costuma-se utilizar as medidas de precisão e cobertura. Como há um *trade-off* entre precisão e cobertura (um algoritmo mais rigoroso nas avaliações positivas terá maior precisão e menor cobertura), a *F-measure* faz uma média harmônica dos dois critérios, possibilitando uma medida completa da qualidade do algoritmo (Crestani, Lalmas, Van Rijsbergen, and Campbell (1998)).

4.3.2 Avaliação do estado emocional dos pacientes

Para que um sistema robótico de fisioterapia seja bem sucedido, não é suficiente que o algoritmo tenha boa precisão e acurácia para julgar os movimentos do paciente. É importante também que a experiência seja agradável para o paciente, o que contribui para um menor absenteísmo (quando pacientes faltam às sessões de tratamento) e maior engajamento do paciente durante a sessão (Maclean, Pound, Wolfe, and Rudd (2000)).

Para avaliar o impacto do tratamento no estado emocional do paciente, foi utilizado um simples questionário baseado em Assumpção Jr, Kuczynski, Sprovieri, and Aranha (2000), pedindo que o paciente dê uma nota de um a quatro ao seu estado emocional naquele instante (nota um corresponde a muito infeliz, dois a pouco infeliz, três a pouco feliz, e quatro a muito feliz). O mesmo questionário foi aplicado antes e depois do tratamento.

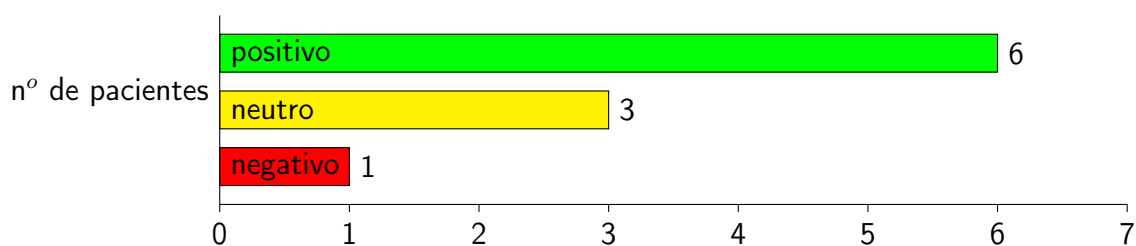


Figura 4.4: Impacto no estado emocional do paciente, comparando-se respostas ao questionário antes e depois do tratamento.

A comparação das respostas antes e depois do tratamento observada na figura 4.4 mostra que o impacto do tratamento no estado emocional dos pacientes foi neutro (três pacientes mantiveram a nota constante), positivo (seis pacientes aumentaram a nota), e em um caso negativo (uma criança de seis anos de idade ficou com medo do robô e não completou o experimento). Isto indica que o tratamento foi bem recebido pelos pacientes, melhorando seu bem-estar e contribuindo para uma sessão de fisioterapia mais produtiva. Esta conclusão é reforçada pela avaliação do estado emocional dos pacientes durante a sessão, discutida na próxima seção.

4.3.3 Reconhecimento de emoções via modelamento geométrico de faces

Para avaliar o estado emocional do paciente durante o tratamento, foi utilizado o sistema de reconhecimento de emoções apresentado na seção 3.5.

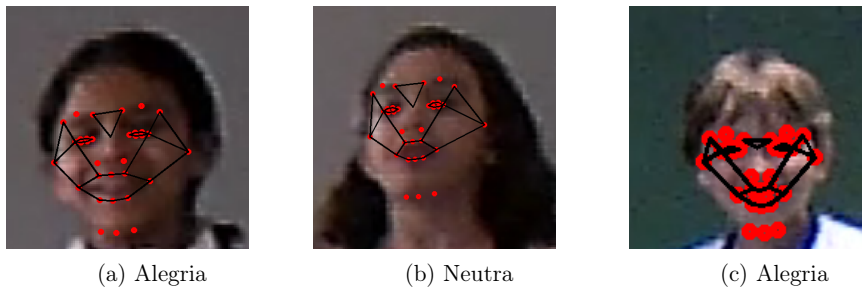


Figura 4.5: Rostos de pacientes mapeados pelo sistema de Libralon and Romero (2013), mostrando emoções de alegria e neutralidade.

A partir de vídeos das sessões, foram capturadas imagens da expressão facial de cada um dos dez pacientes durante o tratamento, com intervalos de trinta segundos entre cada captura, resultando em um total de sessenta e três capturas com resultado válido. Na Figura 4.5 podem vistos resultados da classificação obtida pelo sistema de reconhecimento de gestos correspondentes a três imagens coletadas nos vídeos.

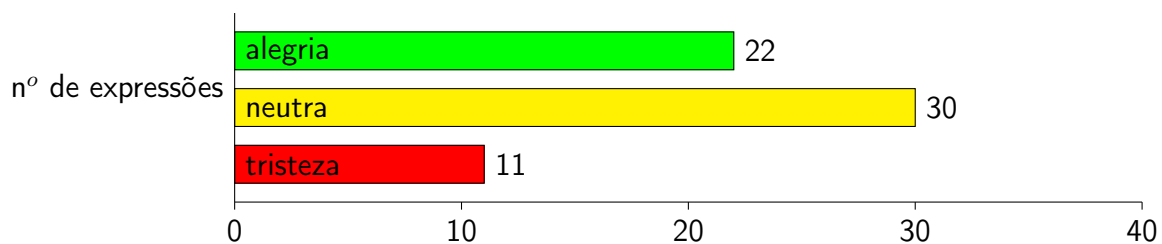


Figura 4.6: Impacto no estado emocional do paciente, comparando-se respostas ao questionário antes e depois do tratamento.

Os resultados mostram expressões faciais neutras, de alegria e de tristeza. Conforme apresentado na figura 4.6, das expressões faciais válidas capturadas, 22 foram de alegria, 30 foram neutras e 11 foram avaliadas como sendo de tristeza.

As expressões de tristeza vieram principalmente da criança de seis anos com paralisia cerebral leve, que chorou de medo do robô NAO e não completou o experimento. Em outras sete ocasiões, uma expressão de esforço foi interpretada como tristeza pelo *software*. Nos outros nove pacientes, obteve-se expressões faciais de alegria, principalmente, quando o robô NAO falava e demonstrava os movimentos. Expressões faciais neutras foram obtidas quando o paciente estava concentrado realizando o movimento. Portanto, considerando expressões faciais de alegria e neutras como esperadas, tem-se uma taxa de aceitação de aproximadamente 83% de satisfação do paciente durante a sessão de fisioterapia com a participação do robô. Isto reforça a conclusão obtida pelos questionários antes e depois do exercício, de que o tratamento foi agradável aos pacientes, melhorando seu bem-estar e contribuindo para uma sessão de fisioterapia mais produtiva e agradável.

4.4 *Considerações Finais*

Neste capítulo foram apresentados o método para modelamento de gestos baseado em máquinas de estados finitos, o sistema integrado para análise e verificação do gesto realizado pelo paciente e também os vários experimentos realizados para verificar o desempenho do sistema proposto. Os resultados obtidos foram discutidos e mostraram que o sistema pode ser uma alternativa no auxílio aos fisioterapeutas.

Conclusão

Neste trabalho foi proposto um sistema integrado, que alia software e um robô humanoide, por meio de uma interface humano/robô, visando o reconhecimento de gestos e correção de exercícios fisioterapêuticos.

Embora o custo do robô NAO, utilizado neste trabalho, seja alto, o que pode inviabilizar a aplicabilidade do sistema integrado proposto, ele se apresenta com uma contribuição inovadora tanto para a área de Fisioterapia quanto para a área de Robótica. Para a área de fisioterapia, apresenta-se como uma proposta alternativa para entretenimento do paciente, enquanto se exercita, tornando mais agradável as repetições dos exercícios. Por outro lado, contribui também para a área de robótica de reabilitação, pois, pelo que é do nosso conhecimento, o robô NAO, apesar de já ter sido utilizado em outras aplicações na área médica, ainda foi utilizado para auxiliar em sessões de fisioterapia.

Além disso, vislumbra-se num futuro próximo, que os custos de robôs deste tipo tendam a se tornar mais baratos, o que poderia reduzir significativamente o custo de tratamentos fisioterapêuticos, permitindo que um único fisioterapeuta supervisione o tratamento concomitante de múltiplos pacientes, cada um sendo guiado individualmente por um robô deste tipo. Por um lado, a redução de custos liberaria recursos médicos para outras áreas e facilitaria que populações carentes tenham acesso aos recursos de saúde. Por outro lado, facilitaria o trabalho de fisioterapeuta durante as sessões. O maior engajamento e diversão proporcionados pela interação com o robô podem também levar a melhores resultados, já que pacientes mais dedicados e concentrados costumam ter sessões mais produtivas.

O uso de máquinas de estados finitos, com cada estado determinado pelo agrupamento de poses por meio da técnica *k-means*, permite que cada exercício seja dividido de forma natural em movimentos parciais, que são demonstrados um por vez ao paciente pelo robô,

e assim cada movimento parcial pode ser corrigido tão logo um erro seja detectado pelo sistema.

A discrepância entre a pose observada e o estado desejado permite que o robô comunique ao paciente uma correção pré-programada adequada à situação. A alternativa mais comum para detecção de gestos seria avaliar o exercício como um todo por meio de um modelo oculto de Markov, para então informar ao paciente se o exercício foi executado corretamente ou não, mas isso tornaria difícil para o paciente saber onde foi seu erro.

Vale observar que o uso de máquina de estados finitos também ajuda a resolver o problema fundamental do reconhecimento de gestos: um dado movimento em geral não é realizado de forma idêntica por diferentes pessoas, ou mesmo pela mesma pessoa em diferentes ocasiões. Sempre existirão diferenças anatômicas, de posição, e de velocidade entre uma execução do gesto e outra. A variância computada em torno do vetor de centroides de articulações que define cada estado provê a tolerância necessária para aceitar diferenças anatômicas e de posição para um dado gesto executado em diferentes condições, enquanto que a tolerância de tempo mínimo e máximo para a transição entre estados provê a tolerância para diferenças de velocidade de execução do gesto.

O sistema integrado proposto foi utilizado em sessões de fisioterapia com vários pacientes, crianças e adultos, que apresentavam algum tipo de problema nas articulações dos membros superiores. Além disso, ele foi bem recebido em uma Clínica de Fisioterapia de São Carlos, tanto pelas fisioterapeutas quanto pelos pacientes. Os resultados obtidos demonstraram o bom desempenho do sistema uma vez que foi observada significativa melhora no estado emocional dos pacientes quando comparados via questionário aplicado antes e depois de cada seção, bem como na avaliação de faces medida durante a realização dos exercícios, indicando que o tratamento foi bem recebido pelos pacientes, melhorando seu bem-estar e contribuindo para uma seção de fisioterapia mais produtiva.

Os resultados demonstraram que sistema robótico desenvolvido obteve bons resultados tanto no reconhecimento e correção de exercícios fisioterapêuticos (com alta precisão e cobertura) quanto na aceitação, engajamento e felicidade dos pacientes (com medidas objetivas mostrando melhora no estado emocional como consequência do tratamento). O uso de um sensor não intrusivo permitiu fácil adaptação dos pacientes e as características amigáveis do robô NAO contribuíram para tornar a experiência divertida para os participantes.

Contudo, nenhum experimento foi realizado com pacientes com problemas nas articulações dos membros inferiores, o que exigiria uma calibração do equilíbrio do robô na execução dos movimentos que deveriam ser replicados pelos pacientes.

Este trabalho gerou o seguinte artigo:

Gambirasio, R.; Romero, R.A.F., "VISUAL GESTURE RECOGNITION FOR MIMICKING AND CORRECTING MOVEMENTS IN ROBOT-GUIDED PHYSIOTHERAPY", que foi submetido ao journal Computer Vision and Image Understanding (CVIU), em 2015.

5.1 *Trabalhos futuros*

Trabalhos futuros ainda precisam avaliar o impacto de longo prazo deste sistema; talvez após passar a novidade do robô os pacientes se cansem e prefiram voltar a interagir somente com o fisioterapeuta humano. Um tratamento de longo prazo com o sistema robótico, comparado com um grupo de controle recebendo o tratamento usual, é necessário para demonstrar se de fato o maior engajamento dos pacientes notado neste trabalho leva a melhores resultados.

O robô NAO também é limitado nos tipos de movimentos humanos que pode reproduzir, seja por limitações de suas articulações ou por dificuldades de equilíbrio, então seria útil adaptar o sistema para uso com outros robôs que permitam uma gama maior de movimentos, desde que se mantenha a apresentação amigável do NAO.

A decisão do número k de estados ideal para a melhor experiência do paciente ainda requer intuição e testes pelo programador do sistema, e varia de acordo com o exercício a ser aprendido. As correções pré-programadas também dependem caso a caso do programador. Embora não leve ainda a automação completa da criação da máquina de estados finitos com base no movimento observado, o uso de *dynamic k-means clustering* pode permitir que o número de estados k seja determinado automaticamente pelo sistema, reduzindo a necessidade de intervenção humana no processo, e é uma extensão natural deste trabalho a ser testada.

Referências Bibliográficas

- Farshid Amirabdollahian, Rui Loureiro, Bart Driessen, and William Harwin. Error correction movement for machine assisted stroke rehabilitation. *Integration of Assistive Technology in the Information Age*, 9:60–65, 2001.
- Francisco B Assumpção Jr, Evelyn Kuczynski, Maria Helena Sprovieri, and Elvira MG Aranha. Escala de avaliação de qualidade de vida. *Arq neuropsiquiatr*, 58(1):119–127, 2000.
- Emilia I Barakova and Tino Lourens. Expressing and interpreting emotional movements in social games with robots. *Personal and Ubiquitous Computing*, 14(5):457–467, 2010.
- Leonard E Baum, Ted Petrie, George Soules, and Norman Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The annals of mathematical statistics*, 41(1):164–171, 1970.
- Antal K Bejczy and Zoltan Szakaly. Universal computer control systems (uccs) for space telerobots. In *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, volume 4, pages 318–324. IEEE, 1987.
- Tanya N Beran, Alex Ramirez-Serrano, Otto G Vanderkooi, and Susan Kuhn. Reducing children’s pain and distress towards flu vaccinations: A novel and effective application of humanoid robotics. *Vaccine*, 31(25):2772–2777, 2013.
- Aaron F Bobick and Andrew D Wilson. A state-based approach to the representation and recognition of gesture. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(12):1325–1337, 1997.
- Ulrich Borgolte, R Hoelper, and Helmut Hoyer. An omnidirectional wheelchair with enhanced comfort features. In *Proceedings International Conference on Rehabilitation Robotics (ICORR’97)*, pages 31–34. nv, 1997.
- Bambi R Brewer, Sharon K McDowell, and Lise C Worthen-Chaudhari. Poststroke upper extremity rehabilitation: a review of robotic systems and clinical results. *Topics in stroke rehabilitation*, 14(6):22–44, 2007.

- Thiago Chaves, Lucas Figueiredo, Alana Da Gama, Cristiano de Araujo, and Veronica Teichrieb. Human body motion and gestures recognition based on checkpoints. In *Virtual and Augmented Reality (SVR), 2012 14th Symposium on*, pages 271–278. IEEE, 2012.
- Gordon Cheng and Yasuo Kuniyoshi. Real-time mimicking of human body motion by a humanoid robot. In *Proceedings of the Sixth International Conference on Intelligent Autonomous Systems (IAS2000), Venice, Italy*, pages 273–280. Citeseer, 2000.
- Carlos Cifuentes, Ariel Braidot, Luis Rodríguez, Melisa Frisoli, Alfonso Santiago, and Anselmo Frizera. Development of a wearable zigbee sensor system for upper limb rehabilitation robotics. In *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS & EMBS International Conference on*, pages 1989–1994. IEEE, 2012.
- Fabio Crestani, Mounia Lalmas, Cornelis J Van Rijsbergen, and Iain Campbell. “is this document relevant?... probably”: a survey of probabilistic models in information retrieval. *ACM Computing Surveys (CSUR)*, 30(4):528–552, 1998.
- Daniel Q Freitas Alana EF Da, Gama Lucas Figueiredo Thiago M Chaves, Déborah Marques-Oliveira, and Veronica Teichrieb Cristiano Araújo. Development and evaluation of a kinect based motor rehabilitation game.
- James Davis and Mubarak Shah. Visual gesture recognition. In *Vision, Image and Signal Processing, IEE Proceedings-*, volume 141, pages 101–106. IET, 1994.
- Paulo G De Barros and Robert W Lindeman. A survey of user interfaces for robot teleoperation. 2008.
- Agnes Delaborde, Marie Tahon, Claude Barras, and Laurence Devillers. A wizard-of-oz game for collecting emotional audio data in a children-robot interaction. In *Proceedings of the International Workshop on Affective-Aware Virtual Agents and Social Robots*, page 5. ACM, 2009.
- Fletcher Dunn and Ian Parberry. *3D math primer for graphics and game development*. CRC Press, 2011.
- Paul Ekman and Wallace V Friesen. *Unmasking the face: A guide to recognizing emotions from facial clues*. Ishk, 2003.
- S Sidney Fels and Geoffrey E Hinton. Glove-talk: A neural network interface between a data-glove and a speech synthesizer. *Neural Networks, IEEE Transactions on*, 4(1): 2–8, 1993.
- M Fengler and WM Cameron. Clinical testing of a low cost robotic arm for the severely physically disabled. In *Proceedings of the First International Workshop on Robotic Applications in Medical and Health Care*, 1988.

- William R Ferrell. Remote manipulation with transmission delay. *Human Factors in Electronics, IEEE Transactions on*, (1):24–32, 1965.
- William R Ferrell and Thomas B Sheridan. Supervisory control of remote manipulation. *Spectrum, IEEE*, 4(10):81–88, 1967.
- Caroline Fu. An independent vocational workstation for a quadriplegic. *Interactive Robotic Aids-One Option for Independent Living: An International Perspective, volume Monograph*, 37:42, 1986.
- Janez Funda and RP Paul. A symbolic teleoperator interface for time-delayed underwater robot manipulation. In *OCEANS'91. Ocean Technologies and Opportunities in the Pacific for the 90's. Proceedings.*, pages 1526–1533. IEEE, 1991.
- Alana Da Gama, Thiago Chaves, Lucas Figueiredo, and Veronica Teichrieb. Guidance and movement correction based on therapeutics movements for motor rehabilitation support systems. In *Virtual and Augmented Reality (SVR), 2012 14th Symposium on*, pages 191–200. IEEE, 2012.
- Rodolphe Gelin, Christophe daAlessandro, Q Anh Le, Olivier Deroo, David Doukhan, Jean-Claude Martin, Catherine Pelachaud, Albert Rilliard, and Sophie Rosset. Towards a storytelling humanoid robot. In *AAAI Fall Symposium Series*, 2010.
- Jennifer Goetz and Sara Kiesler. Cooperation with a robotic assistant. In *CHI'02 Extended Abstracts on Human Factors in Computing Systems*, pages 578–579. ACM, 2002.
- Michael A Goodrich and Alan C Schultz. Human-robot interaction: a survey. *Foundations and trends in human-computer interaction*, 1(3):203–275, 2007.
- Karen Guldberg. Karen guldberg talks about nao. <http://asknao.aldebaran-robotics.com/en/community/blog/invited-writer/karen-guldberg-talks-about-NAO>, 2013. [Online; accessed 31-July-2013].
- Michael Hillman. Rehabilitation robotics from past to present-a historical perspective. In *The 8th International Conference on Rehabilitation Robotics PROCEEDINGS*, pages 1–4, 2003.
- Peter F Hokayem and Mark W Spong. Bilateral teleoperation: An historical survey. *Automatica*, 42(12):2035–2057, 2006.
- Pengyu Hong, Matthew Turk, and Thomas S Huang. Constructing finite state machines for fast gesture recognition. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 3, pages 691–694. IEEE, 2000a.
- Pengyu Hong, Matthew Turk, and Thomas S Huang. Gesture modeling and recognition using finite state machines. In *Automatic face and gesture recognition, 2000. proceedings. fourth ieee international conference on*, pages 410–415. IEEE, 2000b.

- Takashi Imaida, Yasuyoshi Yokokohji, M Oda, T Yoshikawa, et al. Ground-space bilateral teleoperation of ets-vii robot arm by direct bilateral coupling under 7-s time delay condition. *Robotics and Automation, IEEE Transactions on*, 20(3):499–511, 2004.
- Davood Khalili and Michael Zomlefer. An intelligent robotic system for rehabilitation of joints and estimation of body segment parameters. *Biomedical Engineering, IEEE Transactions on*, 35(2):138–146, 1988.
- Y Kim and AM Cook. Manipulation and mobility aids, 1985.
- Barbara Klein, Lone Gaedt, and Glenda Cook. Emotional robots: Principles and experiences with paro in denmark, germany, and the uk. *GeroPsych: The Journal of Gerontopsychology and Geriatric Psychiatry*, 26(2):89, 2013.
- Jonas Koenemann and Maren Bennewitz. Whole-body imitation of human motions with a nao humanoid. In *Human-Robot Interaction (HRI), 2012 7th ACM/IEEE International Conference on*, pages 425–425. IEEE, 2012.
- Hermano I Krebs, Michael Krams, Dimitris K Agrafiotis, Allitia DiBernardo, Juan C Chavez, Gary S Littman, Eric Yang, Geert Byttebier, Laura Dipietro, Avrielle Rykman, et al. Robotic measurement of arm movements after stroke establishes biomarkers of motor recovery. *Stroke*, 45(1):200–204, 2014.
- KA Kristy, SJ Wu, RF Erlandson, P deBear, D Geer, and M Dijkers. A robotic armsmart exercise system’: a rehabilitation therapy modality. In *Engineering in Medicine and Biology Society, 1989. Images of the Twenty-First Century., Proceedings of the Annual International Conference of the IEEE Engineering in*, pages 1504–1505. IEEE, 1989.
- Belinda Lange, Sebastian Koenig, Eric McConnell, C Chang, Rick Juang, Evan Suma, Mark Bolas, and Albert Rizzo. Interactive game-based rehabilitation using the microsoft kinect. In *Virtual Reality Short Papers and Posters (VRW), 2012 IEEE*, pages 171–172. IEEE, 2012.
- Hyeon-Kyu Lee and Jin-Hyung Kim. An hmm-based threshold model approach for gesture recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(10):961–973, 1999.
- Giampaolo L Libralon and Roseli A F Romero. Mapping of facial elements for emotion analysis. In *The Brazilian Conference on Intelligent Systems (BRACIS)*, pages 222–227. IEEE, 2014.
- Giampaolo L Libralon and Roseli AF Romero. Geometrical facial modeling for emotion recognition. In *Neural Networks (IJCNN), The 2013 International Joint Conference on*, pages 1–8. IEEE, 2013.

- S Lichiardopol. A survey on teleoperation. *Dept. Mech. Eng., Dynamics Control Group, Technische Universiteit Eindhoven, Eindhoven, Dept., Mech. Eng., Dyn. Control Group, The Netherlands, Tech. Rep. DCT2007*, 155, 2007.
- Mark A Livingston, Jay Sebastian, Zhuming Ai, and Jonathan W Decker. Performance measurements for the microsoft kinect skeleton. In *Virtual Reality Workshops (VR), 2012 IEEE*, pages 119–120. IEEE, 2012.
- Peter S Lum, Gitendra Uswatte, Edward Taub, Phillip Hardin, and Victor W Mark. A telerehabilitation approach to delivery of constraint-induced movement therapy. *Journal of rehabilitation research and development*, 43(3):391, 2006.
- David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.
- Niall Maclean, Pandora Pound, Charles Wolfe, and Anthony Rudd. Qualitative analysis of stroke patients’ motivation for rehabilitation. *Bmj*, 321(7268):1051–1054, 2000.
- A Madni, Yee-yeen Chu, and E Freedy. Intelligent interface for remote supervision and control of underwater manipulation. In *OCEANS’83, Proceedings*, pages 106–110. IEEE, 1983.
- John Makhoul, Francis Kubala, Richard Schwartz, Ralph Weischedel, et al. Performance measures for information extraction. In *Proceedings of DARPA broadcast news workshop*, pages 249–252, 1999.
- Laura Marchal-Crespo and David J Reinkensmeyer. Review of control strategies for robotic movement training after neurologic injury. *Journal of neuroengineering and rehabilitation*, 6(1):20, 2009.
- CP Mason and E Peizer. Medical manipulator for quadriplegic. In *Proc Int’l Conf. on Telemanipulators for the Physically Handicapped*, 1978.
- Maja Matarić, Adriana Tapus, Carolee Winstein, and Jon Eriksson. Socially assistive robotics for stroke and mild tbi rehabilitation. *Advanced Technologies in Rehabilitation*, 145:249–262, 2009.
- Stephen J McKennah and Shaogang Gong. Gesture recognition for visually mediated interaction using probabilistic event trajectories. 1998.
- Michelle McNickle. Medical robots that could change healthcare. <http://www.informationweek.com/healthcare/mobile-wireless/10-medical-robots-that-could-change-heal/240143983>, 2012. [Online; accessed 31-July-2013].

- Luis Molina-Tanco, JP Bandera, Rebeca Marfil, and F Sandoval. Real-time human motion analysis for human-robot interaction. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 1402–1407. IEEE, 2005.
- Dariush Mozaffarian, Emelia J Benjamin, Alan S Go, Donna K Arnett, Michael J Blaha, Mary Cushman, Sarah de Ferranti, Jean-Pierre Despres, Heather J Fullerton, Virginia J Howard, et al. Heart disease and stroke statistics-2015 update: a report from the american heart association. *Circulation*, 131(4):e29, 2015.
- Shinichiro Nakaoka, Atsushi Nakazawa, Kazuhito Yokoi, Hirohisa Hirukawa, and Katsushi Ikeuchi. Generating whole body motions for a biped humanoid robot from captured human dances. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, volume 3, pages 3905–3910. IEEE, 2003.
- Nils J Nilsson. Shakey the robot. Technical report, DTIC Document, 1984.
- PD Nisbet, JP Odor, and IR Loudon. The call centre smart wheelchair. In *First Int. Conf. on Robotics Apps. in Medical and Healthcare*, 1988.
- Takuichi Nishimura and Ryuichi Oka. Spotting recognition of human gestures from time-varying images. In *Automatic Face and Gesture Recognition, 1996., Proceedings of the Second International Conference on*, pages 318–322. IEEE, 1996.
- Christian Ott, Dongheui Lee, and Yoshihiko Nakamura. Motion capture based human motion recognition and imitation by direct marker control. In *Humanoid Robots, 2008. Humanoids 2008. 8th IEEE-RAS International Conference on*, pages 399–405. IEEE, 2008.
- Marcia Riley, Ales Ude, Keegan Wade, and Christopher G Atkeson. Enabling real-time full-body imitation: A natural way of transferring human movement to humanoids. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, volume 2, pages 2368–2374. IEEE, 2003.
- H Roesler, HJ Kuppers, and E Schmalenbach. The medical manipulator and its adapted environment: A system for the rehabilitation of severely handicapped. In *IRIA Proc. Int. Conf. Telemanipulators for the Physically Handicapped*, pages 73–77, 1978.
- Raquel Ros, Marco Nalin, Rachel Wood, Paul Baxter, Rosemarijn Looije, Yannis Demiris, Tony Belpaeme, Alessio Giusti, and Clara Pozzi. Child-robot interaction in the wild: advice to the aspiring experimenter. In *Proceedings of the 13th international conference on multimodal interfaces*, pages 335–342. ACM, 2011.
- Syamimi Shamsuddin, Hanafiah Yussof, Luthffi Ismail, Fazah Akhtar Hanapiah, Salina Mohamed, Hanizah Ali Piah, and Nur Ismarrubie Zahari. Initial response of autistic children in human-robot interaction therapy with humanoid robot nao. In *Signal*

Processing and its Applications (CSPA), 2012 IEEE 8th International Colloquium on, pages 188–193. IEEE, 2012.

Thomas B Sheridan and William R Ferrell. Remote manipulative control with transmission delay. *Human Factors in Electronics, IEEE Transactions on*, (1):25–29, 1963.

PC Shor, PS Lum, CG Burgar, HFM Van der Loos, M Majmundar, and R Yap. The effect of robotic-aided therapy on upper extremity joint passive range of motion and pain. *Assistive Technology Research Series*, 9:79–83, 2001.

Steven B Skaar and Carl F Ruoff. *Teleoperation and robotics in space*. Number 161. Aiaa, 1994.

Halit Bener Suay and Sonia Chernova. Humanoid robot control using depth camera. In *Human-Robot Interaction (HRI), 2011 6th ACM/IEEE International Conference on*, pages 401–401. IEEE, 2011.

Susumu Tachi, Kazuo Tanie, Kiyoshi Komoriya, and Minoru Abe. Electrocutaneous communication in a guide dog robot (meldog). 1985.

Edward Taub, Gitendra Uswatte, Danna Kay King, David Morris, Jean E Crago, and Anjan Chatterjee. A placebo-controlled trial of constraint-induced movement therapy for upper extremity after stroke. *Stroke*, 37(4):1045–1049, 2006.

Robert W Teasell and Lalit Kalra. What’s new in stroke rehabilitation back to basics. *Stroke*, 36(2):215–217, 2005.

Mike Topping. Handy 1, a robotic aid to independence for severely disabled people. In *7th Int’l Conf. on Rehabilitation Robotics*, pages 142–147, 2001.

Andrew Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, IEEE Transactions on*, 13(2):260–269, 1967.

Stefan Waldherr, Roseli Romero, and Sebastian Thrun. A gesture based interface for human-robot interaction. *Autonomous Robots*, 9(2):151–173, 2000.

Daniel E Whitney. State space models of remote manipulation tasks. *Automatic Control, IEEE Transactions on*, 14(6):617–623, 1969.

Junji Yamato, Jun Ohya, and Kenichiro Ishii. Recognizing human action in time-sequential images using hidden markov model. In *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR’92., 1992 IEEE Computer Society Conference on*, pages 379–385. IEEE, 1992.

Jie Yang and Yangsheng Xu. Hidden markov model for gesture recognition. Technical report, DTIC Document, 1994.

DR Yoerger and J-JE Slotine. Supervisory control architecture for underwater teleoperation. In *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, volume 4, pages 2068–2073. IEEE, 1987.

Fernando Zuher and Roseli Romero. Recognition of human motions for imitation and control of a humanoid robot. In *Robotics Symposium and Latin American Robotics Symposium (SBR-LARS), 2012 Brazilian*, pages 190–195. IEEE, 2012.