

DESENVOLVIMENTO DE UM SISTEMA DE VISÃO PARA ROBÔ HUMANOIDE

Renato Jacobiski Fusco ¹; Alessandra Dutra Coelho ²; Wânderson de Oliveira Assis ²

¹ Aluno de Iniciação Científica da Escola de Engenharia Mauá (EEM-CEUN-IMT);

² Professor(a) da Escola de Engenharia Mauá (EEM-CEUN-IMT).

Resumo. *Este trabalho tem o objetivo principal de aprimorar os hardwares e softwares de um robô humanoide autônomo que se movimenta com o auxílio de uma câmera embarcada de forma semelhante ao sistema de visão humana. O desenvolvimento do trabalho será baseado em um projeto anterior e consiste no aperfeiçoamento de um sistema de sensoramento baseado em uma câmera embarcada, com movimentação pan-tilt, que permita ao robô movimentação, reconhecimento do ambiente e decidir qual será sua ação para alcançar seus objetivos. Para isso, o sistema de movimentação do robô será baseado em processamento de imagens e uso de sensores como acelerômetros e giroscópios que deverão dar estabilidade e movimentação precisa ao robô. Apesar de ter um objetivo simples de uma competição de futebol de robôs (a Robocup), o conceito de movimentação de um robô autônomo utilizando câmeras embarcadas é de grande importância e centro de muitas pesquisas na atualidade. Este sistema permite a proximidade com um sistema conhecidamente eficiente de sensoramento, a visão humana.*

Introdução

Robôs autônomos são definidos como um conjunto de sistemas que realizam uma ou mais tarefas sem um explícito ou contínuo controle humano e devido ao avanço tecnológico sua presença está cada vez mais importante na sociedade. Desde pequenos insetos robóticos até veículos não tripulados para exploração de áreas perigosas, a automação influencia grande inovações tecnológicas. Medicina, segurança, fábricas e operações de procura em ambientes distantes ou inóspitos são algumas das áreas que tiveram um grande progresso por causa da robótica.

Uma das formas de criar um controle de um robô autônomo é construindo um sistema de visão. Neste tipo de modelo, ele capta informações do meio externo, faz um processamento com esses dados e se adapta para uma melhor resposta à situação. No projeto anterior foi utilizado uma placa arduino nano e uma câmera CMUCam4 para fazer a captura e o processamento da imagem, o que demonstrou algumas limitações.

No exemplo utilizado neste trabalho será implementada uma câmera USB em um robô cujo objetivo é localizar uma pequena bola laranja, fazendo com que o robô se movimente até ela e a chute para marcar pontos.

A visão computacional é uma ciência recente que iniciou seus projetos em meados dos anos 70. Neste período, destacam-se os projetos Winston, P.H., The Psychology of Computer Vision e Minsky, Marvin, A Framework for Representing Knowledge, ambos de 1975. Com o

desenvolvimento de novos trabalhos, os pesquisadores compreenderam a complexidade da área e descobriram limitações, como a falta de informações e modelos que representassem a forma como as imagens são interpretadas no cérebro humano.

Atualmente existem tecnologias mais aprimoradas que permitem uma excelente captura e processamento de imagens. A manipulação de sinais multi-dimensionais está se desenvolvendo diariamente para aprimorar a qualidade de fotos tiradas por satélites, métodos de exploração espacial e etc. No caso deste trabalho, a melhoria da obtenção e análise de imagens é de extrema importância para melhorar o desempenho do robô humanoide.

Objetivos e Metas

O objetivo deste trabalho é a melhoria do sistema de visão utilizado por um robô humanoide que será usado na Robocup (futebol de robôs). O atual sistema deste modelo possui muitas limitações, tanto em hardware como em software. A mudança da placa do microprocessador e da câmera embarcada por uma USB com software OpenCV conseguirá trazer resultados otimizados e mais precisos para o *tracking* de objetos específicos.

O sistema embarcado utiliza uma lente de captura e um microcontrolador para o processamento de imagens e para a automação da resposta dada pelo robô. A câmera utilizada será simples e barata, aplicável a um sistema embarcado, e estará “embarcada”, ou seja, instalada em um robô móvel (*embedded system*). No robô humanoide escolhido deve ser introduzido um sistema eletrônico programável que permita a interação com o sistema de captura. O sistema de automação embarcado deve, além disso, permitir a interação com o meio a partir do processamento das imagens de forma a controlar o movimento do robô.

A câmera entrará em comunicação com uma placa Beaglebone Black que vai processar a imagem desta e, observando padrões RGB e quantidade de “pixels” confiáveis, obterá a posição do humanoide em relação a um objeto. Como o robô é bípede, seu equilíbrio ao se movimentar é instável. Operações com giroscópio e acelerômetro são essenciais para assegurar o estado em que o humanoide se encontra servindo como sensores de posição.

A aquisição da imagem com a placa Beaglebone Black traz como vantagens uma programação com linguagem de alto nível e uma melhor análise de dados, além de sua entrada HDMI permitir a visualização em tempo real da imagem “vista” pelo robô. Por ser de uma geração mais nova, a placa permite a comunicação com modelos mais avançados de câmeras com qualidade e resolução melhores. Pelo fato de seu processamento ser mais rápido e mais completo, as ações do humanoide também serão mais velozes e precisas, pontos chave para a otimização de sua função.

Material e Métodos

Materiais utilizados

O robô humanoide utilizado é o Robonova-1 (figura 1) produzido pela *HiTec*. O robô contém 16 servomotores do tipo HSR-8498 e uma placa de controle MRC-3024 (figura 2). A placa

MRC-3024 contém um microcontrolador do tipo ATMEL ATMEGA 128 e é programada por uma interface em linguagem específica do Robonova-1, o *RoboBasic*, a qual é uma linguagem simples. Por essa linguagem ser básica e apresentar diversas limitações, foi preciso utilizar outros recursos para melhorar o controle do robô.



Figura 1 -Robonova-1.

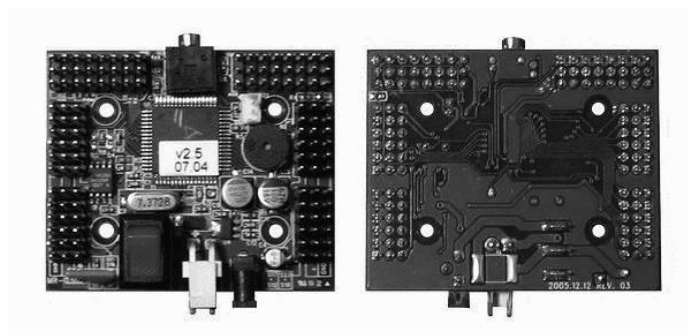


Figura 2- MRC-3024.

O sistema de visão adotado no projeto é composto por uma câmera embarcada (Webcam), a qual pode ser vista na figura 3 e utiliza a biblioteca openCV. Foi implantado um sistema de movimentação com 2 graus de liberdade (sistema *pan-tilt*), como pode ser visto na figura 4. Este sistema é utilizado para movimentação nos três eixos de um espaço, o que permite a liberdade de movimentação necessária para a ação desejada. A câmera Webcam Microsoft LifeCam VX-800 utilizada entrará em comunicação com uma placa de processamento para esta processar a imagem capturada e, observando padrões RGB e quantidade de “pixels” confiáveis, obterá uma posição do humanoide em relação à bola laranja. As características desta câmera podem ser vistas na tabela 1.

Tabela 1 - Informações técnicas da Webcam - Microsoft LifeCam VX-800.

Sistemas Operacionais:	<ul style="list-style-type: none"> • Microsoft Windows® 8.1, Windows 8, Windows RT 8.1, Windows 7 • Microsoft OS X v10.7-10.9 • Android 3.2 and 4.2
Características de imagem:	Sensor CMOS VGA sensor technology
Resolução:	<ul style="list-style-type: none"> • Vídeo movimentado: 0.31 megapixel (640 x 480 pixels) • Imagem parada: 0.31 megapixel (640 x 480 pixels) sem interpolação
Campo de visão:	59° diagonal



Figura 3- Webcam - Microsoft LifeCam VX-800.



Figura 4- Sistema Pan-Tilt com servomotores.

O sistema de controle foi definido a partir da necessidade de obter maior nível de inteligência do robô, podendo haver uma melhor programação e controle das atividades

necessárias. O robô apresenta uma placa MRC-3024 que possui um microcontrolador ATMEL ATMEGA 128 programada com uma linguagem específica para o Robonova-1 chamada *ROBOBASIC*. Essa linguagem tem grandes limitações, como a não existência de variáveis tipo indexada e a chamada de estruturas condicionais serem limitadas. Por essas limitações, foram realizadas pesquisas para a substituição da placa de controle por uma que tivesse uma programação mais avançada. O desenvolvimento de uma nova placa de controle foi tema de pesquisas anteriores, mas houve problemas na utilização, o que levou a busca de novos métodos de controle.

No projeto anterior, foi utilizada a placa Arduino no sistema de controle já existente, o que trouxe grandes benefícios em programação e com baixo custo. Porém, este método apresentou diversas limitações técnicas e físicas. A câmera previamente utilizada não era a melhor opção em relação à qualidade da imagem e sua utilização deixava o controle menos prático. A escolha por uma nova placa de processamento otimizou o padrão de processamento e aprimorou o sistema embarcado. A placa retirou da MRC-3024 a necessidade de decisão de movimentos, que era bem limitada, devendo manter apenas as rotinas de movimentação do Robonova-1. As rotinas de movimentação da placa MRC-3024 apresentam importantes características, as quais não seriam possíveis serem reproduzidas no Beaglebone Black (figura 5) como o Grupo de Movimentos, que tem a função de garantir que todos os servomotores terminem um movimento ao mesmo tempo. Também o controle de posições dos servomotores, o que garante a correta posição do robô a certo comando. Outro ponto importante para se manter a placa MRC-3024 em conjunto com o Beaglebone Black é a grande quantidade de servomotores necessários para a movimentação do robô. A placa implementada não suporta 16 servomotores e não tem a capacidade de suprir internamente a quantidade de corrente necessária para movimentar esses servos.

As características do modelo da placa podem ser vistas na Tabela 2.

Tabela 2- Informações técnicas do Beaglebone Black.

Processador	1GHz TI Sitara AM3358BZCZ100 ARM® Cortex A8, 2000 MIPS
Memória Flash	4GB eMMC 8-bit
RAM	512 MB DDR3L @ 800 MHz
Armazenamento	2 GB on-board eMMC, MicroSD
Conectores de vídeo	1 Micro-HDMI
Resoluções	1280×1024 (5:4), 1024×768 (4:3), 1280×720 (16:9), 1440×900 (16:10) all at 16 bit
Áudio	Stereo pelo HDMI
Sistemas operacionais	Angstrom (Default), Ubuntu, Android, ArchLinux, Gentoo, Minix, RISC OS, outros...
Consumo de Energia	210-460 mA @ 5V sobre condições variáveis
Dimensões	3.4" × 2.1"
Capacidade GPIO	65 Pinos
Periféricos	1 USB Host, 1 Mini-USB Client, 1 10/100 Mbps Ethernet

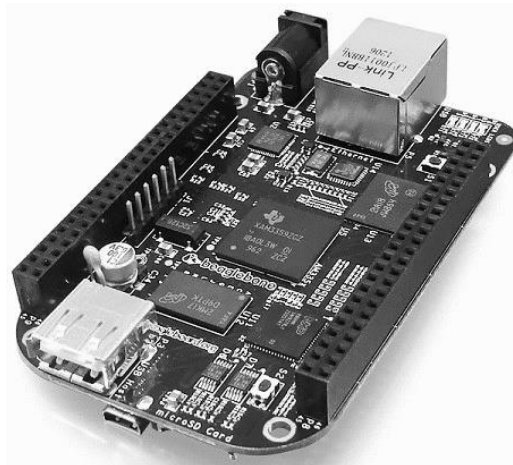


Figura 5- Beaglebone Black.

Pelo fato do Beaglebone Black ser considerado um computador portátil, a programação é feita na própria placa. Utilizando um HUB USB, conecta-se um teclado e um mouse em sua entrada USB e a interface pode ser vista ao acoplar um monitor em sua entrada HDMI. O Beaglebone possui também algumas bibliotecas disponíveis para diferentes casos, o que torna mais rápida a criação de códigos mais complexos. A linguagem de programação utilizada foi o Python, modelo utilizado pelo sistema operacional vigente na placa: Linux (Angstrom) e foram utilizados alguns programas em C para a integração com o software OpenCV.

Os sensores adicionais usados no robô são os acelerômetros e giroscópios. Os acelerômetros medem as acelerações em um eixo específico, o que permite obter informações sobre impactos, forças ou até sobre a queda livre de um objeto. Através dele podem-se analisar as forças que estão agindo sobre o robô e obter um melhor equilíbrio quando houver um movimento não desejado. Há diversos tipos de acelerômetros existentes, mas os mais utilizados são os acelerômetros de material piezoelétrico e os de capacitância. Os acelerômetros de material piezoelétrico são formados por uma massa presente no interior de uma caixa e um material piezoelétrico. A característica de um material piezoelétrico é a de que quando ele é comprimido, surge uma voltagem elétrica e através dessa voltagem pode-se analisar quanto a massa comprime esse material através do seu movimento. A massa se movimenta no interior da caixa e quando há uma aceleração, a massa tende a manter seu estado inicial o que, relativamente, seria o mesmo a dizer que a massa acelera em direção oposta ao da caixa. Essa inércia gera uma força, já que $F=m.a$, onde “m” é a massa e “a” é a aceleração. Já nos acelerômetros de capacitância existe outra propriedade envolvida que é a distância entre duas placas inversamente polarizadas. Quanto mais próxima uma placa estiver da outra, menor será a diferença de potencial elétrico entre os terminais negativos e positivos do capacitor. Quando se utiliza um acelerômetro de capacitância uma das placas é presa a uma mola e a uma massa e quando há aceleração a mola é esticada, o que diminui a distância em relação à outra placa e gera uma variação na capacitância produzida por elas. Essa diferença é analisada e pode-se observar a aceleração realizada.

Os giroscópios são muito importantes, principalmente em sistemas de navegação de aeronaves, navios, piloto automático, voos espaciais e etc. Ele se baseia em dois conceitos importantes da física que são a inércia e a precessão, a qual consiste na mudança do eixo de rotação de um objeto. O giroscópio é constituído basicamente de um rotor suspenso com um eixo em

rotação. Essa roda mantém uma posição fixa mesmo após ser forçada em uma determinada direção não paralela. Através desse sistema pode se observar a direção do objeto em relação ao rotor, o que permite a correta orientação do corpo. Para se obter as orientações em um espaço tridimensional é necessário que haja 3 giroscópios em posições perpendiculares. Já nos equipamentos eletrônicos, o giroscópio usado é da categoria MEMS (*Microelectromechanical systems*) ou até mesmo NEMS (*Nanoelectromechanical systems*), que são dispositivos extremamente pequenos (nanotecnologia) e que são muito utilizados em equipamentos de eletrônica embarcada, como automóveis e aviões. Os giroscópios do tipo MEMS utilizam o conceito de Pêndulo de Foucault (experiência baseada na movimentação de um pêndulo sendo modificada pela rotação da terra) e usam elementos vibratórios.

Em conjunto, o giroscópio e o acelerômetro constituem a base do IMU (*Inercial Measurement Unit*), o qual é um equipamento eletrônico capaz de medir a aceleração, orientação e direção do movimento. Podem ser integrados GPS, magnetômetros e até tubos de Pitot, para um completo controle de movimento no espaço. No Robonova-1, o giroscópio e o acelerômetro serão responsáveis pelo equilíbrio do robô e a correta percepção do seu estado em relação ao desejado. Outro aspecto importante é perceber quando o robô está caído e como ele está caído. Através dos dois sensores poderemos determinar sua posição e agir para levantá-lo de forma adequada.

Projeto e testes

O controle será baseado em etapas de processamento nas quais são necessárias comunicações bilaterais. Abaixo está o esquema simplificado do sistema de controle através da figura 6.

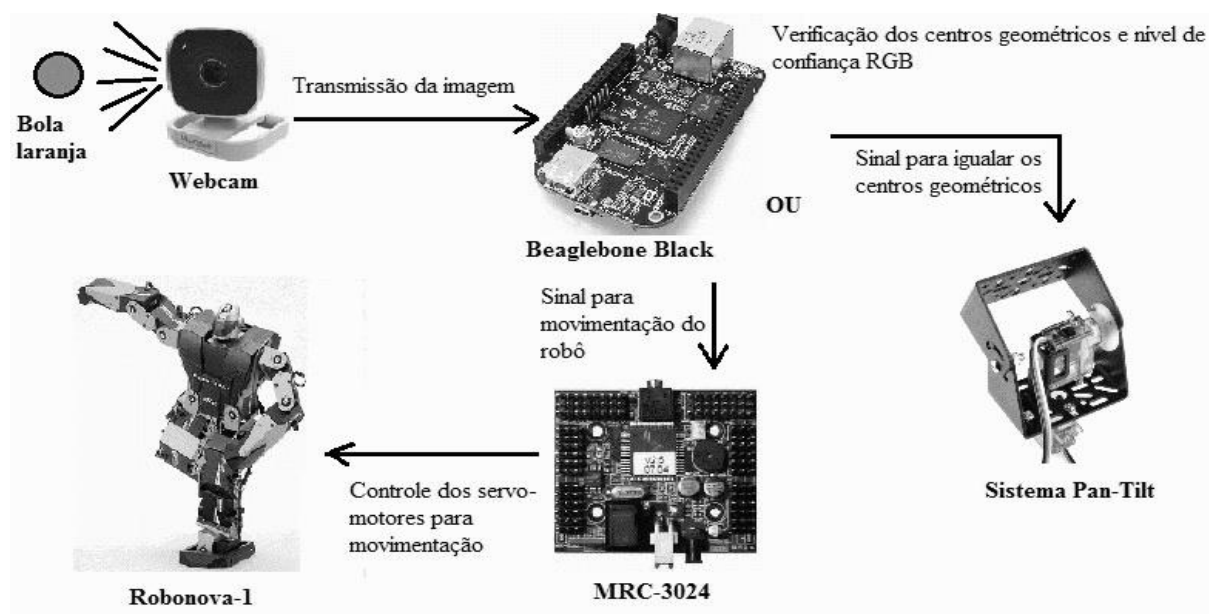


Figura 6 – Esquema simplificado de controle do robô.

A Beaglebone Black é responsável por todo o processamento das imagens capturadas pela câmera e envio de dados sobre o posicionamento do objeto desejado. Através desses dados este realiza o ajuste do sistema pan-tilt até que o centro geométrico da imagem fique exatamente alinhado com o centro da câmera. Isso é possível através de um algoritmo de controle proporcional-derivativo (PD) desenvolvido na placa que faz um movimento estável ao seguir um objeto. O mesmo algoritmo é utilizado tanto para o movimento de “Pan” (rotação no eixo y) quanto para o

movimento de “Tilt” (rotação no eixo x). Quando a câmera não consegue mais observar o objeto, esta volta à posição inicial e começa a procurá-lo. Se a bola não foi encontrada, é enviado um código para o robô girar para o lado até achar o objeto.

O sistema de decisão de comando baseia-se em regras de movimentação definidas. Quando o programa decide uma ação, a placa envia um byte para MRC-3024 que representa a ação desejada. Esse código é representado pelo código ASCII de 0 a 8, como visto a seguir pela tabela 3.

Tabela 3 – Códigos de movimentação.

Ação	ASCII	Decimal	Binário
Posição inicial	0	48	0011 0000
Andar para Frente	1	49	0011 0001
Andar para Trás	2	50	0011 0010
Virar para Direita	3	51	0011 0011
Virar para Esquerda	4	52	0011 0100
Passo para Direita	5	53	0011 0101
Passo para Esquerda	6	54	0011 0110
Levanta (Costas)	7	55	0011 0111
Levanta (Frente)	8	56	0011 1000

Os códigos são recebidos pela MRC-3024 e esta controla os servo-motores para fazerem o movimento definido. O robô executa o movimento uma única vez e envia de volta um sinal informando que a ação acabou e que precisa de novas informações. Isso possibilita um melhor controle e estabilidade física do robô e impede ações não desejadas. Os conjuntos de movimentos presentes na placa MRC-3024 vêm programados de fábrica pela *Hitec*, o que facilita o trabalho de movimentação já que deve ser definido o ângulo de cada motor para cada tipo de deslocamento. Para desenvolver um sistema correto de movimentação para o robô deveria ser feito um modelo completo de todo seu sistema dinâmico, resultando em uma pesquisa mais complexa.

Resultados e Discussão

Os novos equipamentos demonstraram um grande avanço nas funcionalidades do Robonova-1, melhorando suas respostas e a qualidade da imagem. A alteração da câmera embarcada e o desenvolvimento do novo sistema de controle aprimoraram o controle e as ações, observando um grande aumento nas possibilidades do seu uso.

Apesar das melhorias, houve problemas devido à quantidade de equipamentos adicionados na estrutura do robô. O peso da Webcam influenciou no centro de gravidade do robô, o que gerou instabilidade em certos movimentos. Este deve se movimentar de uma maneira mais lenta para não ocorrer quedas e danificá-lo. Há também a necessidade futura de proteções para as placas eletrônicas da parte de trás do Robonova-1 pois com a adição do Beaglebone Black a proteção original não pode mais ser utilizada.

A nova câmera embarcada mostrou melhorias para captar imagens da bola a maiores distâncias. Ainda existe o problema de luminosidade do ambiente, mas com a aquisição da nova placa de controle podem-se utilizar câmeras com melhores resoluções, permitindo um reconhecimento aprimorado de imagens.

O sistema de controle mostrou eficiência na transmissão de dados. Foi observada uma certa dificuldade para rodar softwares e reproduzir as funções certas, mas apesar disso a placa conseguiu identificar o objeto e realizou o processamento como esperado.

Há a necessidade futura de haver uma integração dos sensores giroscópios e acelerômetros na estrutura do robô. Ainda não houve a integração dos sensores no robô e nem o correto aproveitamento dos resultados para uma melhor movimentação. Os dois sensores possuem papel fundamental para se ter uma movimentação estável, mas não foi possível implementar no atual projeto.

Os testes com o sistema Pan-Tilt ainda não foram realizados com os novos equipamentos, mas no projeto anterior (Arduino Nano + CMUCam4) a integração deste modelo com o sistema funcionou corretamente, obtendo velocidades altas ao perseguir o objeto em movimento.

Conclusões

O projeto apresentou resultados positivos e conseguiu o seu objetivo de aprimorar o sistema de visão do robô humanoide, desenvolvendo um sistema mais eficiente de controle e com uma melhor resolução da imagem.

A nova forma de controle se mostrou muito confiável e com grande potencial de melhorias pela sua capacidade de programação e de sensoriamento. A mudança de linguagem de programação e a maior memória disponível tornam-se importantes ferramentas para o desenvolvimento de um controle de qualidade.

Há diversos pontos importantes a serem melhorados na estrutura do robô, mas houve um grande avanço nas capacidades do Robonova-1. Existem algumas possíveis melhorias em relação a qualidade da imagem, o aprimoramento do controle do robô e até mesmo novos meios de visão para um melhor controle que podem ser analisadas em futuras pesquisas.

Através do desenvolvimento, foi de suma importância a pesquisa de rastreamento de objetos. Existem diversos projetos baseados em *Object Tracking* para Beaglebone Black que utilizam OpenCV, contendo variadas aplicações. Quanto melhor o processamento da imagem e percepção do ambiente, melhor será a qualidade de informações disponíveis para um processamento eficiente.

O projeto teve algumas limitações mas, apesar disso, o objetivo de obter uma melhoria no sistema de visão com um custo relativamente baixo e com boa eficiência foi realizado. Por isso, o projeto aprimorou as possibilidades de estudos futuros com sistemas de visão embarcada e o conhecimento adquirido através desta pesquisa pode ser levado a áreas importantes na robótica autônoma com uma certa facilidade de implementação.

Referências Bibliográficas

KIKUCHI, D. Y. “Sistema de controle servo visual de uma câmera pan-tilt com rastreamento de uma região de referência”, Tese (Mestrado) – Escola Politécnica, Universidade de São Paulo, São Paulo, SP, 2007.

GONZALES, R. C., WOODS, R. E. (2000) Processamento de Imagens Digitais. Edgard Blücher Ltda.

ROWE, A. (2003) ROBONOVA-I – 16 Servo Edutainment Robot Box Kit – Introduction Manual. Edited by Charles Rosenberg and Illah Nourbakhsh, Carnegie Mellon University.

Assis, W. O.; Genova, W. J.; Gomes, M. M.; Coelho, A. D. (2007) Aplicação de Tecnologias de Controle de Processos Contínuos e Processamento de Imagens no Futebol de Robôs. *Anais do SBAI 2007 – Simpósio Brasileiro de Automação Inteligentes*, Florianópolis, SC, Brasil.

Jácomo, J. E. A. (2001) Desenvolvimento de um Robô Autônomo Móvel Versátil utilizando Arquitetura Subsumption. Universidade Estadual de Campinas; Faculdade de Engenharia Mecânica; Departamento de Mecânica Computacional, Campinas, SP.

BALLET, R. ; COELHO, Alessandra Dutra ; ASSIS, W. O. . Integrando Tecnologias em Sistema de Controle de Movimentação de Robô Humanóide com Câmera Embarcada. 4. Seminário Mauá de Iniciação Científica, São Caetano do Sul, SP, 2012.

<http://pedrogarcia12av1.wordpress.com/about/robos-autonomos/>

<http://gizmodo.uol.com.br/video-robos-autonomos/>

http://computervision.wikia.com/wiki/Computer_Vision

<http://computacao grafica.ic.uff.br/transparenciasvol2cap3.pdf>

<http://www.eie.polyu.edu.hk/~enyhchan/imagef.pdf>

<http://www.cs.dartmouth.edu/farid/downloads/tutorials/fip.pdf>

<ftp://qiftp.tudelft.nl/DIPimage/docs/FIP.pdf>

<http://www.dpi.inpe.br/>

<http://sergioprado.org/beaglebone/>

<http://beagleboard.org/>

http://elinux.org/ECE597_Opencv_on_the_BeagleBoard

<http://derekmolloy.ie/beaglebone/>

<http://webpages.shepherd.edu/ccrite03/Project%20Website/index.html>

<http://www.inape.org.br/colunas/fisica-conceito-historia/pendulo-foucault>