

UNIVERSIDADE SAGRADO CORAÇÃO

RAFAEL MORAES CANARIN

**SISTEMA PARA CONTROLE DE UNIDADE ROBÓTICA
POR MEIO DA REALIDADE AUMENTADA**

**BAURU
2012**

RAFAEL MORAES CANARIN

**SISTEMA PARA CONTROLE DE UNIDADE ROBÓTICA
POR MEIO DA REALIDADE AUMENTADA**

Trabalho de Conclusão de Curso apresentado ao Centro de Ciências Exatas e Sociais Aplicadas como parte dos requisitos para obtenção do título de bacharel em Ciência da Computação, sob orientação do Prof.Ms. Wiliam Carlos Galvão.

**BAURU
2012**

C213s	<p data-bbox="548 1249 852 1281">Canarin, Rafael Moraes</p> <p data-bbox="548 1312 1274 1407">Sistema para controle de unidade robótica por meio da realidade aumentada / Rafael Moraes Canarin -- 2012.</p> <p data-bbox="597 1417 690 1449">58f.: il.</p> <p data-bbox="597 1480 1153 1512">Orientador: Prof. Ms. Wiliam Carlos Galvão</p> <p data-bbox="548 1543 1274 1638">Trabalho de Conclusão de Curso (Graduação em Ciência da Computação - Bacharelado) – Universidade Sagrado Coração – Bauru – SP.</p> <p data-bbox="548 1669 1274 1764">1. Java. 2. Robótica. 3. Realidade aumentada. 4. NyARToolKit. 5. Arduino. I. Galvão, Wiliam Carlos. II. Título.</p>
-------	---

RAFAEL MORAES CANARIN

**SISTEMA PARA CONTROLE DE UNIDADE ROBÓTICA POR MEIO DA
REALIDADE AUMENTADA**

Trabalho de Conclusão de Curso apresentado ao Centro de Ciências Exatas e Sociais Aplicadas como parte dos requisitos para obtenção do título de bacharel em Ciência da Computação, sob orientação do Prof. Ms. Wiliam Carlos Galvão.

Banca examinadora:

Prof. Dr. Elvio Gilberto da Silva
Universidade Sagrado Coração

Prof. Esp. Henrique Pachioni Martins
Universidade Sagrado Coração

Prof. Ms. Wiliam Carlos Galvão
Universidade Sagrado Coração

Bauru, 11 de Junho de 2012.

AGRADECIMENTOS

Agradeço aos meus pais, que sempre me ajudam com os seus conselhos e fazem o melhor para alcançar os meus objetivos na vida.

Aos professores, que estão dispostos a ajudar e ensinar no que for preciso.

Ao meu orientador, Prof. Ms. Wiliam Carlos Galvão, pela sua disponibilidade, vontade, atenção e conhecimento.

E por fim, a minha principal motivadora, minha namorada e futura mulher, Carolina de Carli, que me apoia e me motiva a melhorar sempre e realizar meus objetivos.

“Ser o homem mais rico do cemitério não me interessa. Ir para a cama à noite dizendo que fiz alguma coisa maravilhosa é o que importa para mim.” (Steve Jobs).

RESUMO

Com as novas áreas de trabalho e situações de alta periculosidade, surge há necessidade em que o homem tenha mais precisão e controle no que faz, e com isso a automação veio para substituir as tarefas do ser humano. Tendo em vista isso e a ausência de projetos correlacionados esse trabalho teve como objetivo desenvolver um sistema na linguagem de programação *Java* para controlar uma unidade robótica por meio da Realidade Aumentada, podendo aumentar a facilidade e precisão de cada movimento do robô, por meio da utilização da biblioteca *NyARToolKit*. A utilização de um protótipo eletrônico *open-source* (código aberto), neste caso o *Arduino*, que nada mais é que um *hardware* que permite o envio de comando por linguagem *Java* que possibilitou a interação do computador com controle de motores. Além do *Arduino*, foram necessários outros componentes que acompanham junto com o *kit*, como dois motores de movimento (controle das rodas), motor do movimento da cabeça, banco de baterias de 9,6V. O sistema possui três menus com alguns submenus e quatro botões para facilitar a interação. Um dos botões captura a imagem da *webcam*, e quando um dos cinco marcadores forem reconhecidos, executa um código para a movimentação do *Arduino*, que pode ser para qualquer direção exceto diagonal. A comunicação é feita pela porta *USB (Universal Serial Bus)*, mas que é criada uma porta serial virtual, para a transmissão dos códigos ao *Arduino*. Podendo assim qualquer usuário controlar ou manusear qualquer tipo de unidade robótica com um simples movimento.

Palavras Chave: *Java*, Robótica, Realidade aumentada, *NyARToolKit*, *Arduino*.

ABSTRACT

With new areas of work and highly dangerous situations, there is a need that man has more precision and control at it, and with it came the automation to replace tasks of human. Considering this and the absence of related projects that work aimed at developing a system in the Java programming language to control a robotic unit through Augmented Reality, may increase the ease and precision of each movement of the robot through the use of NyARToolKit library. The use of a prototype electronic open-source, in this case the Arduino, which is nothing more than a hardware that allows the sending of command by java language that allowed the interaction of computer-controlled engines. Arduino addition, it was necessary other components that accompany with the kit, such as two motors motion (control wheel), the head movement motor, the battery bank of 9.6 v. The system has three menus and submenus with some four buttons for easy interaction. A button captures the image of the webcam, and when one of the five markers are recognized, executes code for handling the Arduino, which can be in any direction except diagonally. Communication is done via USB (Universal Serial Bus), but that creates a virtual serial port, for transmitting the code to the Arduino. So any user can control or handle any type of robotic unit with a simple movement.

Keywords: Java, Robotic, Augmented reality, NyARToolKit, Arduino.

LISTA DE ILUSTRAÇÕES

FIGURA 1 - INTERAÇÃO DE <i>HARDWARE</i> E <i>SOFTWARE</i>	10
FIGURA 2 - APLICAÇÃO DA REALIDADE AUMENTADA.	11
FIGURA 3 - <i>ARDUINO</i>	12
FIGURA 4 - EXEMPLO DE UMA <i>WEBCAM</i>	13
FIGURA 5 - EXEMPLO DE UMA UNIDADE ROBÓTICA.	14
FIGURA 6 - EXEMPLO DE UMA UNIDADE ROBÓTICA.	14
FIGURA 7 - EXEMPLO DE AMBIENTE COLABORATIVO DE R.A.	17
FIGURA 8 - FASES DA REALIDADE MISTURADA.	18
FIGURA 9 - EXEMPLO DE UMA APLICAÇÃO DA R.A.	18
FIGURA 10 - FUNCIONAMENTO DO <i>NYARTOOLKIT</i>	19
FIGURA 11 - EXEMPLO DE UM MARCADOR.	20
FIGURA 12 - EXEMPLO DE UM <i>QR CODE</i>	21
FIGURA 13 - EXEMPLOS DE <i>ARDUINO</i>	23
FIGURA 14 - COMPONENTES DO <i>ARDUINO</i>	24
FIGURA 15 - <i>IDE</i> DO <i>ARDUINO</i>	25
FIGURA 16 - EXEMPLO DE DISPOSITIVOS DE ENTRADA.	25
FIGURA 17 - EXEMPLO DE UMA <i>WEBCAM</i> DE <i>NOTEBOOK</i>	26
FIGURA 18 - <i>RXTX</i> NO <i>ARDUINO</i>	27
FIGURA 19 - EXEMPLO DE UMA PORTA PARALELA.	27
FIGURA 20 - PORTA SERIAL.	28
FIGURA 21 - FLUXOGRAMA DAS ETAPAS DE FUNCIONAMENTO DO SISTEMA	31
FIGURA 22 - HIERARQUIA ESTRUTURAL DE FUNCIONAMENTO E UTILIZAÇÃO DO SISTEMA.	33
FIGURA 23 - MODELO DO SISTEMA.	35
FIGURA 24 - BOTÃO DA <i>WEBCAM</i>	35
FIGURA 25 - MENU ARQUIVO.	35
FIGURA 26 - JANELA DA <i>WEBCAM</i>	36
FIGURA 27 - <i>SERVOS</i>	36
FIGURA 28 - PONTE <i>HH</i>	37
FIGURA 29 - MOTOR <i>DC</i>	37
FIGURA 30 - PROTÓTIPO DO CARRO.	38

FIGURA 31 - <i>ARTOOLKIT MAKER GENERATOR</i>	39
FIGURA 32 - LINHA DE CÓDIGO QUE CARREGA MARCADOR.....	39
FIGURA 33 - ENVIA LETRA AO <i>ARDUINO</i>	39
FIGURA 34 - CÓDIGO DE ATIVAÇÃO DOS MOVIMENTOS.....	40

SUMÁRIO

1	INTRODUÇÃO	10
1.1	JUSTIFICATIVA	13
2	OBJETIVOS	15
2.1	OBJETIVO GERAL.....	15
2.2	OBJETIVOS ESPECÍFICOS.....	15
3	REALIDADE AUMENTADA	16
3.1	NYARTOOLKIT	19
3.2	MARCADORES.....	20
3.2.1	QrCode	20
3.3	JAVA.....	21
3.3.1	NetBeans	22
3.3.2	Java3D	22
3.3.3	Java Media Framework	22
3.4	ROBÓTICA.....	23
3.4.1	Arduino	23
3.5	DISPOSITIVOS DE ENTRADA	25
3.6	INTERFACES DE COMUNICAÇÃO	26
3.6.1	Rxtx	27
3.6.2	Porta Paralela	27
3.6.3	Porta Serial	28
3.6.4	Porta USB	28
4	METODOLOGIA	30
4.1	MATERIAIS	30
4.2	DESENVOLVIMENTO.....	30
5	CONSIDERAÇÕES FINAIS	41
6	APRIMORIAMENTO FUTUROS	42
	REFERÊNCIAS	43
	APÊNDICE A - MANUAL DO SISTEMA	48
	APÊNDICE B - ARTIGO	60

1 INTRODUÇÃO

A cada dia surgem novas pesquisas sobre como melhorar e facilitar as operações que exijam maior grau de eficiência e precisão nos procedimentos, tal questão nota-se nas máquinas das indústrias, montadoras, metalúrgicas e robôs que desenvolvem tarefas com funções similares a de um ser humano, assim como nos locais em que a acessibilidade para humanos são quase impossíveis ou de alto risco.

Diante deste contexto, ressalta-se a necessidade em adotar-se automação com o uso de dispositivos (*hardwares*) e os *softwares* para a realização de determinadas tarefas, Figura 1.



Figura 1 - Interação de *hardware* e *software*.
Fonte: Heimlich (2008).

De acordo com Ribeiro (2001), automação substitui as tarefas de um humano ou animal por máquina.

“Automação é a operação de máquina ou de sistema automaticamente ou por controle remoto, com a mínima interferência do operador humano.” (RIBEIRO, 2001, p.1).

Hardware, que é a parte física, e o *software*, que pode ser o sistema operacional ou programas, apesar de serem distintos são também equivalentes em sua importância, segundo Morimoto (2007).

Diante deste contexto, questiona-se por que não utilizar a Realidade Aumentada a qual atualmente é a revolução na área digital.

A Realidade Aumentada é a criação de objetos virtuais 3D em um computador, na sua maioria, como mostra a Figura 2.



Figura 2 - Aplicação da realidade aumentada.
Fonte: Elaborado pelo autor.

A definição de inserção de objetos virtuais em ambiente real pode ser tratar da Realidade Aumentada, mostrando através de um dispositivo tecnológico em tempo real, utilizando a interface do ambiente real, para manipular e visualizar objetos virtuais e reais, de acordo com Kirner (2008).

Segundo Fonseca (2010), o conceito de *software* e *hardware* faz parte do *Arduino* e tem código aberto. O surgimento do *Arduino* na Itália em 2005 teve como objetivo desenvolver dispositivo para controlar protótipos desenvolvidos com menos custo do que outros no mercado.

Mas para a realização do desenvolvimento de uma Realidade Aumentada é necessário algumas ferramentas específicas.

Foi realizado nesse projeto, o desenvolvimento de um sistema que faz uma interação da Realidade Aumentada com um dispositivo, no caso o *Arduino* como ilustrado na Figura 3.

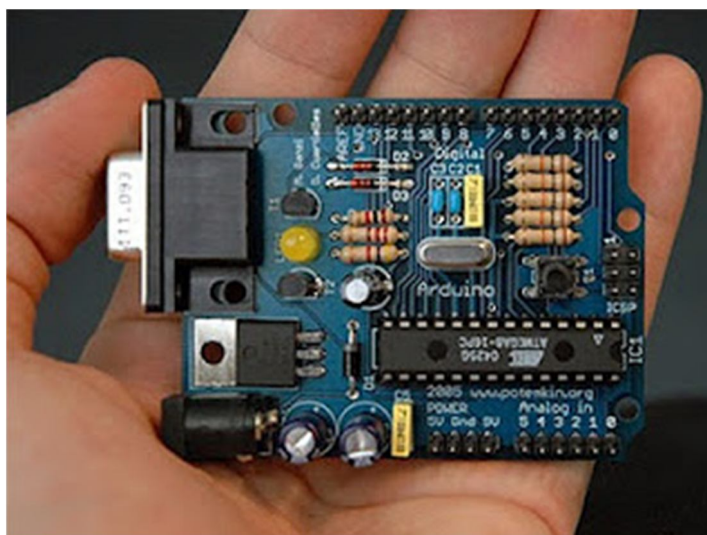


Figura 3 - *Arduino*.
Fonte: Silva (2010).

1.1 JUSTIFICATIVA

Com o vasto crescimento na área da tecnologia, torna-se primordial ao profissional atualizar-se continuamente assim como trabalhar efetivamente com a tecnologia de modo que se possa acompanhar seu desenvolvimento sem desatualizar-se.

Foi levado em consideração o fato da ausência encontrada sobre o tema abordado neste trabalho e que pode ser de grande relevância em termo científico.

A *webcam* (Figura 4) é um dispositivo tecnológico onde captura a imagem e transfere quase que instantaneamente para o computador.



Figura 4 - Exemplo de uma *webcam*.
Fonte: Carlos (2007).

A Realidade Aumentada é uma “técnica” que uni o real com o virtual através da utilização de um marcador e uma *webcam* ou de uma câmera de celular.

Segundo Ramires (2011), a Realidade Aumentada é uma técnica onde se modifica a ideia que uma pessoa tem sobre a realidade. Agindo reciprocamente envolvendo o mundo virtual e a realidade.

Ressalta-se que tendo o controle das unidades robóticas, pode-se evitar acidentes que exijam a presença de um ser humano. Como exemplo prático a Figura 5 demonstra um homem desativando explosivos.



Figura 5 - Exemplo de uma unidade robótica.
Fonte: ARTEFATOS... (2009).

A Figura 6 demonstra o trabalho em uma indústria automotiva em que se faz necessário o manuseio do homem nos equipamentos.

Podendo assim ter precisões e resultados melhores.



Figura 6 - Exemplo de uma unidade robótica.
Fonte: Gasparin (2011).

Diante desse contexto pode-se dizer que temos grandes idéias para ajudar ou solucionar alguns problemas que envolva deficientes.

2 OBJETIVOS

2.1 OBJETIVO GERAL

Desenvolver um sistema em *Java* para controlar uma unidade robótica por meio da Realidade Aumentada.

2.2 OBJETIVOS ESPECÍFICOS

- Fazer uma ampla revisão da bibliografia para contextualização do assunto.
- Realizar um estudo específico da linguagem de programação *Java* para sua aplicação na construção do sistema para o controle da unidade robótica.
- Implementar a tecnologia de Realidade Aumentada no controle da unidade robótica.

3 REALIDADE AUMENTADA

Para a realização desse projeto é necessário o conhecimento sobre a Realidade Aumentada, onde segundo Azuma (1993; FEINER, 1993; Bajura, 1995; Boman, 1995; Kirner; Tori, 2004 apud Zorzal; Kirner, 2005), a Realidade Aumentada pode ser definida como uma tecnologia onde se aumenta ou incrementa a visão sobre o que as pessoas têm do mundo real com a adição de objetos virtuais, usando técnicas de visão por computação gráfica/Realidade Virtual e computador, conseguindo a sobreposições de objetos virtuais no ambiente real.

De acordo com Milgran (1994 apud KIRNER; SISCOOTTO, 2007), é uma combinação de mundos virtuais e reais em algum local da realidade/virtualidade contínua, que conecta ambientes reais a virtuais.

É um sistema onde o computador gera objetos virtuais para serem adicionados em ambientes reais, dando uma ilusão de estarem no mesmo espaço, com as seguintes características: combinação de objeto virtual e real em um ambiente real, execução em tempo real, alinha objetos virtuais e reais entre si e aplica-se em todos os sentidos humanos, conforme Azuma (2001 apud KIRNER; SISCOOTTO, 2007).

Apesar de poder sobrepor objetos virtuais no ambiente real, a Realidade Aumentada também permite o manuseio com as mãos nos objetos, oferecendo aos usuários uma interação atrativa e motivador no ambiente, segundo Billinghurst (2001; Kirner, 2004; Santin, 2004; Zhou, 2004 apud Zorzal; Kirner, 2005).

De acordo com a definição de Azuma (2001 apud Filippo et al., 2007) um sistema de Realidade Aumentada possui as características: objetos virtuais e reais em um ambiente real; ordenam em linha os objetos virtuais e reais uns aos outros, sendo que a Realidade Aumentada trabalha com a manipulação de objetos virtuais, e interagindo no ambiente, havendo um suporte à cooperação, como mostra a Figura 7.



Figura 7 - Exemplo de ambiente colaborativo de R.A.
Fonte: Filippo (2007).

A realidade aumentada é quando se possui a maior parte do mundo real e, virtualidade aumentada quando contrário, ou seja, a maior parte ambiente virtual, conforme Kirner e Tori (2004).

Segundo Zorzal, Buccioli e Kirner (2005), a Realidade Aumentada tem vantagem sobre a virtualidade aumentada, por trazer objetos virtuais ao ambiente real, onde as pessoas possuem uma percepção melhor sobre a situação, funcionamento das regras físicas e o envolvimento com o mundo.

Tomando cuidado com a diferenciação da Realidade Aumentada com a virtualidade aumentada, onde uma é mais forte no mundo real e a outra no ambiente virtual.

A Figura 8 demonstra o supracitado.



Figura 8 - Fases da realidade misturada.
Fonte: Zorzal, Buccioli e Kirner (2005).

Segundo Tori, Kirner e Siscoutto (2006), a realidade aumentada foi muito disseminada, e em muitas ocasiões usada no lugar da realidade misturada.

São as valorizações, perfeição do ambiente real com objetos virtuais por meio de algum dispositivo tecnológico em tempo real, conforme Kirner e Siscoutto (2007).

Segue um exemplo de uma aplicação da Realidade Aumentada, tendo objetos virtuais (carro e o vaso) em um ambiente real (Figura 9).



Figura 9 - Exemplo de uma aplicação da R.A.
Fonte: Kirner (2007).

A próxima etapa para aumentar o realismo e a interação do usuário, é a hiper-realidade, que é uma combinação de elementos e comportamentos.

De acordo com Kirner e Siscoutto (2007) a hiper-realidade é a aptidão de forma natural para acesso do usuário das combinações de realidade virtual, realidade física, inteligência artificial e inteligência humana.

Na realidade virtual ou na realidade aumentada, possui um rastreamento do usuário, seus membros, para ter maior flexibilidade no controle dos ambientes.

Segundo Kirner (2007), o rastreamento tem a função de identificar a posição da mão, cabeça, do usuário, como uma placa.

A realidade virtual e aumentada são áreas recentes que oferece melhores condições de interação com aplicações computacionais, proporcionando interações naturais e potencialização de suas capacidades aos usuários, conforme Kirner e Siscoutto (2007).

3.1 NYARTOOLKIT

O *NyARToolKit* é uma ferramenta criada para atender aos programadores em *Java* para aplicação na Realidade Aumentada.

Com a utilização da ferramenta é possível criar aplicações com as imagens capturadas pela *webcam* de forma simplificada e eficaz, oferecendo muitas variedades para o desenvolvimento de aplicações de Realidade Aumentada e também a Realidade Virtual.

A imagem a seguir (Figura 10) esquematiza os passos do funcionamento do *NyARToolKit* até a projeção dos objetos 3D.

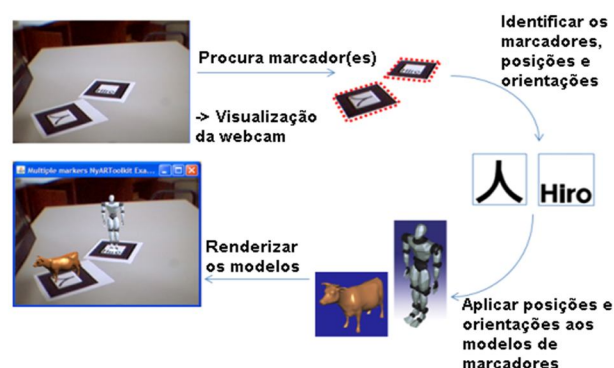


Figura 10 - Funcionamento do *NyARToolKit*.
Fonte: Elaborado pelo autor.

É uma biblioteca que tem capacidade de identificar vários marcadores ao mesmo tempo e projetar seus respectivos objetos.

Ambos, *NyARToolKit* e *ARToolKit*, baseiam-se em identificar certos marcadores, que estão devidamente cadastrados no *software*, podendo também posteriormente criar o seu próprio marcador.

A biblioteca tem suporte para quase todos os sistemas operacionais até mesmo no *Android*, desenvolvido pelo *Google*, utilizado em *smartphones* e *tablets*.

3.2 MARCADORES

Os marcadores (Figura 11), ou também conhecidos como placas com marcas fiduciais, são imagens que podem ser impressas em folhas sulfites (do tipo A4).

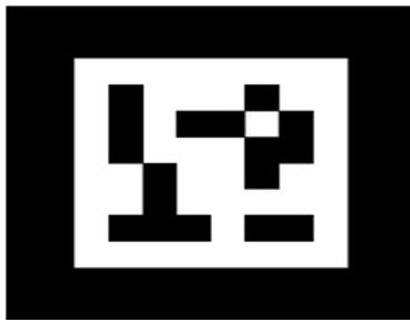


Figura 11 - Exemplo de um marcador.
Fonte: Magalhaes (2009).

“Os marcadores funcionam como monitores virtuais, com a vantagem de poderem ser rearmados livremente na mesa de trabalho e de serem carregados por usuários móveis.” (FILIPPO et al., 2007, p.183).

3.2.1 QRCode

O *Qrcode* (*Quick Response Code*), Figura 12, é um marcador criado para aplicativos de celular que contém uma imagem com informações codificadas.



Figura 12 - Exemplo de um *Qrcode*.
Fonte: Normandia (2010).

Segundo Normandia (2010) para realizar a leitura do *Qrcode* pode se utilizar um leitor específico, celular com câmera ou *smartphone* que rode *Symbian OS*, *Android* ou *iOS*.

3.3 JAVA

O *Java* é uma linguagem de programação de alto nível, orientada a objetos e multiplataforma.

Segundo Morimoto (2005), o *Java* é uma linguagem de programação multiplataforma, com sintaxe similar até certa parte com *C++*, logo com bibliotecas diferentes.

Uma característica da linguagem de programação *Java* é o *Java Native Interface*, que oferece o reaproveitamento de códigos nativos (outras linguagens) dentro dele, segundo Brega et. al. (2008).

De acordo com Furgeri (2010), a programação orientada a objetos é adotada como padrão de mercado e muitas outras linguagens foram aprimoradas para implementar essa forma de trabalho, já o *Java* nasceu assim.

3.3.1 NetBeans

É uma *IDE (Integrated Development Environment)*, ou ambiente integrado de desenvolvimento onde serve como apoio a programadores.

A *IDE NetBeans* é um ambiente de desenvolvimento multiplataforma, uma ferramenta que auxilia programadores a escrever, compilar, debugar e instalar aplicações, foi arquitetada em forma de uma estrutura reutilizável que visa simplificar o desenvolvimento e aumentar a produtividade pois reúne em uma única aplicação todas estas funcionalidades. Totalmente escrita em *Java*, mas que pode suportar qualquer outra linguagem de programação ou linguagem que desenvolva com *Swing*, algumas das linguagens que o *NetBeans* suporta são o *C*, *C++*, *Ruby*, *PHP*, *XML* e linguagens *HTML*. (REDAÇÃO..., 2008).

3.3.2 Java3D

Java3D é uma *API, Application Programming Interface*, ou seja, é uma interface de aplicação na programação que basicamente auxilia no desenvolvimento de sistemas gráficos tridimensionais, baseada nas bibliotecas gráficas *OpenGL* e *DirectX*.

“Possui construtores de alto nível que permitem a criação e manipulação de objetos geométricos, especificados em um universo virtual.” (MANSSOUR, 2003, p. 72).

Com essa *API* é possível também aplicar sons, multimídia, efeito de luz e dentre outros, para aumentar o realismo.

O *Java3D* foi liberado pela *Sun*, a mais de uma década.

Segundo Manssour (2003), a *API* fornece um grande volume de classes para a especificação, posicionamento e a visualização.

3.3.3 Java Media Framework

Se trata de uma *API* desenvolvida pela *Sun* para a linguagem de programação *Java*, que possibilita a utilização e manipulação do áudio, vídeo e outros tipos de mídia.

Segundo Horstmann (2002 apud CAMPOS, 2010; ANDRADE, 2010), é uma coleção de classes que possibilitam a captura e visualização de multimídias.

“O *JMF* é um *framework* da *Sun* (desenvolvido em parceria com a *Intel* e *IBM*) que permite manipular recursos multimídia em aplicações *Java*.” (JORGE, c2012).

Conforme Jorge (c2012) para utilizar os serviços de vídeo ou áudio do *Java Media Framework* são necessários dispositivos de entrada (microfones, câmeras,...) e/ou dispositivos de saída (monitor e caixa de som).

3.4 ROBÓTICA

São desenvolvidos com a finalidade de realizar as mesmas tarefas que um ser humano possa ou não fazer.

Segundo Pires (2002) o termo robô vem do Checo *robot* que significa trabalho, e foi usado pela primeira vez por Karel Capek, e suas máquinas eram de aspecto humano.

A robótica é uma máquina com multifunções e reprogramáveis que podem realizar serviços dos seres humanos, possuindo uma habilidade de identificar alterações nas condições e restrições imposta na tarefa e pelo meio envolvente, decidir qual a ação a fazer e planejá-la, conforme Pires (2002).

3.4.1 Arduino

O *Arduino* (Figura 13) é uma plataforma de prototipagem eletrônica *open-source* que se baseia em *hardware* e *software* flexível de fácil uso.

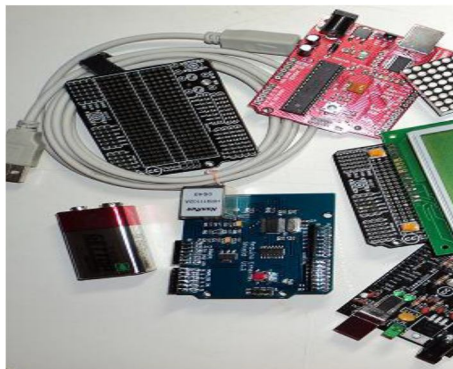


Figura 13 - Exemplos de *Arduino*.
Fonte: Elaborado pelo autor.

De acordo com Fonseca (2010), o *Arduino* é uma plataforma de computação física (sistemas digitais ligados a sensores e atuadores), baseada em uma simples placa de entrada/saída micro controlada e desenvolvida sobre uma biblioteca em C/C++.

A (Figura 14), mostra um esquema do *Arduino*, onde se faz a comunicação serial e onde recebe e envia os dados.

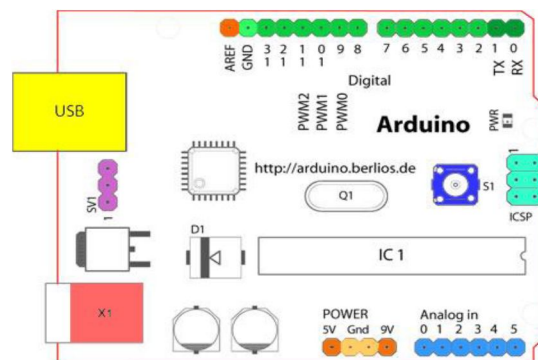


Figura 14 - Componentes do Arduino.
Fonte: *INTRODUCCIÓN...* (c2012).

Arduino é capaz de identificar variáveis e enviar sinais elétricos para a tal ação.

Segundo Fonseca (2010) é um kit de desenvolvimento capaz de transformar em sinais elétricos as variáveis no ambiente, através de sensores, e controlar outro elemento eletroeletrônico conectado ao terminal de saída.

Conforme Fonseca (2010) foi projetado com o objetivo de ser simples para fácil compreensão, programação e na sua aplicação, uma grande vantagem dele é o fato de ser multiplataforma, podendo configurar no *Windows*, *Linux* e *Mac OS*.

A linguagem de programação utilizada é o C/C++, muito conhecida entre os programadores.

É necessário um *Sketch* (Figura 15), como é conhecido, ou simplesmente uma *IDE* para poder programá-lo, atualmente se encontra na versão 0022.

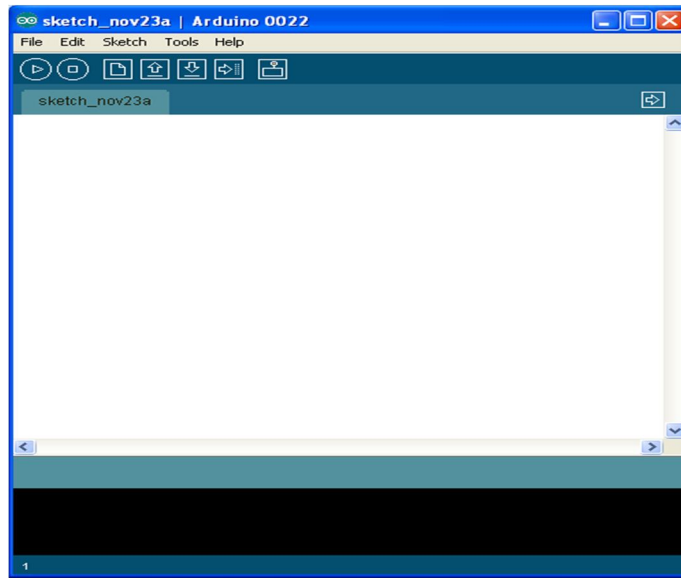


Figura 15 - IDE do Arduino.
Fonte: Elaborado pelo autor.

3.5 DISPOSITIVOS DE ENTRADA

Um dispositivo de entrada (Figura 16) permite a comunicação no sentido do utilizador para o computador.



Figura 16 - Exemplo de dispositivos de entrada.
Fonte: Carolina (2010).

É a forma da comunicação do usuário com o computador.

São dispositivos que enviam dados analógicos ao computador para processamento. Exemplos: Teclado, *mouse*, caneta ótica, *scanner*, câmera de vídeo.

“As funções desses dispositivos são coletar informações e introduzir as informações na máquina, converter informações do homem para a máquina e vice-versa, e recuperar informações dos dispositivos de armazenamento.” (MANO, 2011).

Nesse projeto será utilizado um dispositivo de entrada, uma *webcam* (Figura 17), onde se captura imagens do alvo ou local.



Figura 17 - Exemplo de uma *webcam* de *notebook*.
Fonte: Elaborado pelo autor.

Segundo Rodrigues (2010) a qualidade da *webcam* é o resultado da taxa de quadros (expressa em *FPS* (quadros por segundo), sendo entre 15 a 30 *FPS* em *webcams* comuns) e da resolução da câmera, mas as lentes também são importantes.

3.6 INTERFACES DE COMUNICAÇÃO

O papel da interface de comunicação é justamente de comunicar equipamentos com as máquinas pelas suas pinagens.

Interfaces de comunicação são padrões lógicos e físicos em relação a como são ligados e transmitidos os sinais entre equipamentos e meios de comunicação. Fisicamente constitui de conectores que têm uma pinagem toda específica onde são transmitidas as informações e também são emitidos dados de controle, tais como início de uma transmissão, finalização, confirmação de recebimento, etc. O tipo de interface geralmente está vinculado à velocidade da rede de comunicação. (ARNEIRO, 2006)

3.6.1 Rxtx

Rxtx é uma *API* desenvolvida baseada na *Javacomm* para *Java*, em que realiza a comunicação serial ou paralela e até mesmo pela *USB*.

Foi desenvolvida para facilitar ao desenvolvedor a enviar dados pela porta em questão.

Encontra-se na versão 2.2, mas está em fase de teste.

Na imagem (Figura 18) a seguir, mostra o *Rxtx* no *Arduino*.

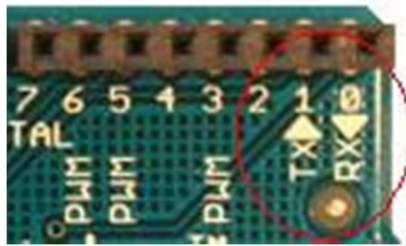


Figura 18 - *Rxtx* no *Arduino*.
Fonte: Elaborado pelo autor.

3.6.2 Porta Paralela

A porta paralela (Figura 19) é por onde se comunica o computador a um dispositivo.

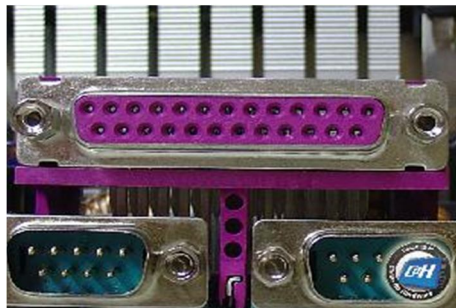


Figura 19 - Exemplo de uma porta paralela.
Fonte: Torres (2005).

De acordo com Pimenta e Faria (2003), a porta é muito utilizada pelo fato de enviar vários *bits* de dados através de oito condutores paralelos distintos em simultâneo, ou seja, pode enviar um *byte* de uma só vez, e o processo recebe o nome de Transmissões Paralelas.

3.6.3 Porta Serial

A porta serial comparada com a paralela é parecida, e às vezes confundida visualmente, pois a o que difere é o seu tamanho que é menor, e conseqüentemente menos pinagens (9 pinos).

A forma de transmissão de *bits* é seqüencial, ou seja, são enviados um de cada vez os dados, assim possuindo uma menor velocidade, porém pode percorrer uma grande distância.

Segundo Morimoto (2007), o nome “serial” é devido à forma de transmissão de bit, um de cada vez.

A figura 20, a seguir, é um exemplo de uma porta serial.

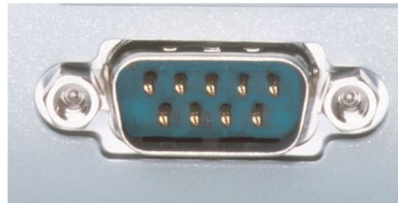


Figura 20 - Porta Serial.
Fonte: Boeira (2011).

3.6.4 Porta USB

A sigla *USB* significa *Universal Serial Bus*, ou seja, Barramento Serial Universal. Uma porta que é utilizada para vários tipos de periféricos, como o *mouse*, teclado, impressora e câmera de vídeo.

“O *USB* é um novo padrão para a conexão de periféricos externos. Suas principais armas são a facilidade de uso e a possibilidade de se conectar vários periféricos a uma única porta *USB*.” (MORIMOTO, 2002)

Segundo Morimoto (2002), o *USB* permite uma taxa de transmissão de *12mbps*, sendo 100 vezes mais rápida que uma porta serial e um pouco mais da paralela.

A atualização do *driver* de um periférico novo que está sendo colocado no computador é feita automaticamente, só algumas vezes é necessário fazer manualmente.

É possível retirar o cabo *USB* em qualquer momento que o sistema operacional reconhece a ação.

Conforme Morimoto (2002) é possível conectar e desconectar periféricos mesmo com o computador ligado, sendo apenas necessário o *driver* do dispositivo.

4 METODOLOGIA

4.1 MATERIAIS

Após estudos e pesquisas em livros e internet, foi feito um planejamento dos procedimentos para o projeto.

Inicialmente foi utilizado um computador *Intel Pentium 4*, 2400 MHz (18 x 133), placa mãe *Asus P4S533-X* (6PCI, 1AGP, 2 SDR DIMM, 2 DDR DIMM, Audio, LAN), Memória 1024 MB (DDR SDRAM) e placa de vídeo *GeForceFX 5200* com um sistema operacional *Windows XP Professional SP3*.

Depois foi utilizado um *notebook HP G42 Intel Pentium (R) Dual-Core T4500 2.3 GHz*, uma placa mãe da *HP* modelo 1484, 2GB de memória DDR3 e com uma placa de vídeo *Mobile Intel(R) 4 Series Express Chipset Family* e sistemas operacionais *Windows XP Professional SP3* e *Windows Seven Home Basic SP3*.

Os *softwares NyARToolKit* (kit de ferramenta de Realidade Aumentada) para as programações necessárias e o *IDE NetBeans*.

A biblioteca *RXTX* por ser um dos mais conhecidos e utilizados, para fazer a transmissão de dados no sistema pela porta *USB*.

Para o desenvolvimento do sistema foi utilizado a linguagem de programação *Java*, por se tratar de uma linguagem de programação diferenciada: Orientada a objeto e ser multiplataforma (pode ser executada em qualquer máquina).

Uma plataforma de computação física, o *Arduino*.

Um cabo *USB*.

Uma *webcam* da *HP*.

4.2 DESENVOLVIMENTO

Inicialmente foram instalados os *softwares*, como o *NetBeans* na versão 7.1 para programação em *Java*, *ARToolKit Maker Generator* para criação dos marcadores, *Java3D* para os objetos 3D, *Java Media Frame* para a captura da *webcam* e as

bibliotecas *NyARToolKit* para o desenvolvimento da Realidade Aumentada e o *RXTX* para a comunicação com a porta serial.

Após isso foi necessário reiniciar o computador, para efetivação da configuração. Segue abaixo o ciclo do sistema (Figura 21).

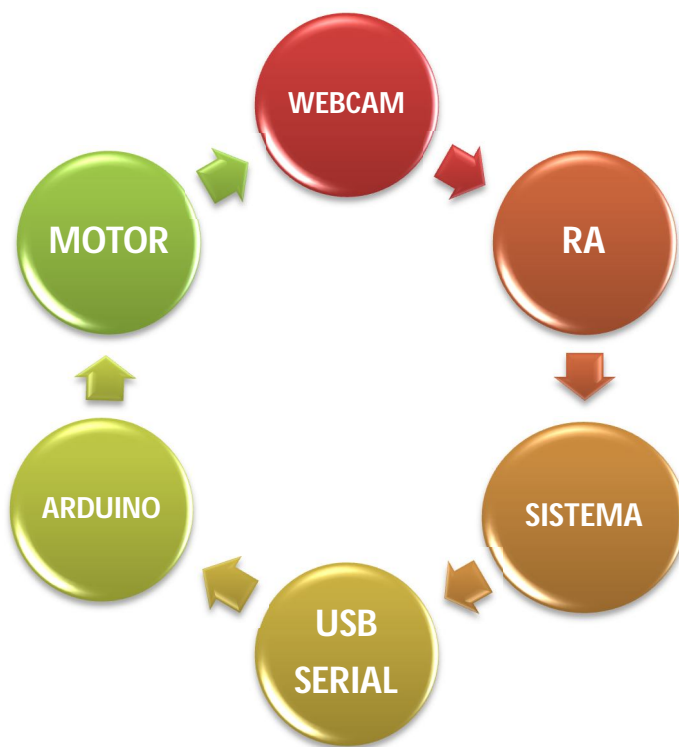
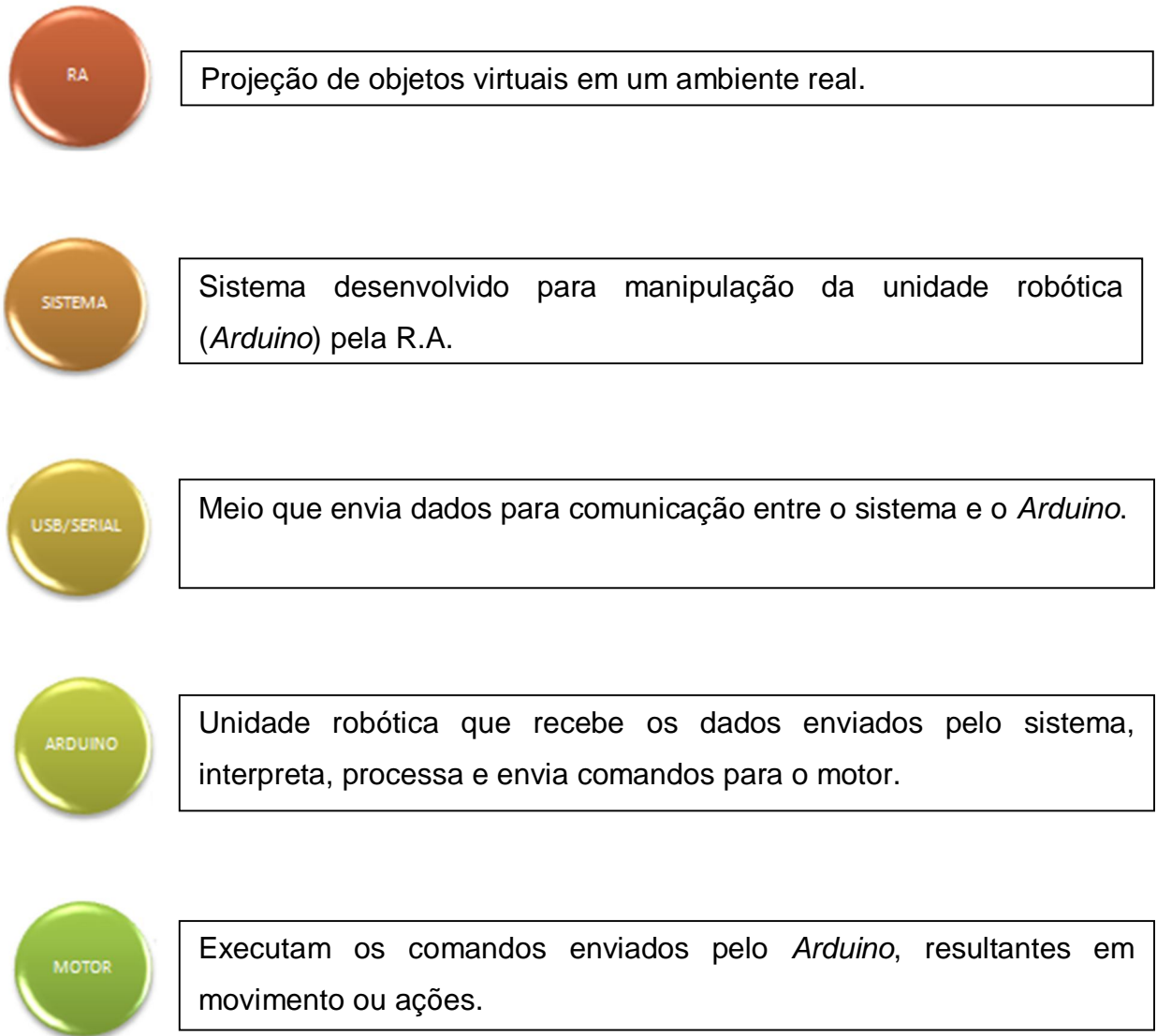


Figura 21 - Fluxograma das etapas de funcionamento do sistema
Fonte: Elaborado pelo autor.

Descrição das etapas de funcionamento do sistema.



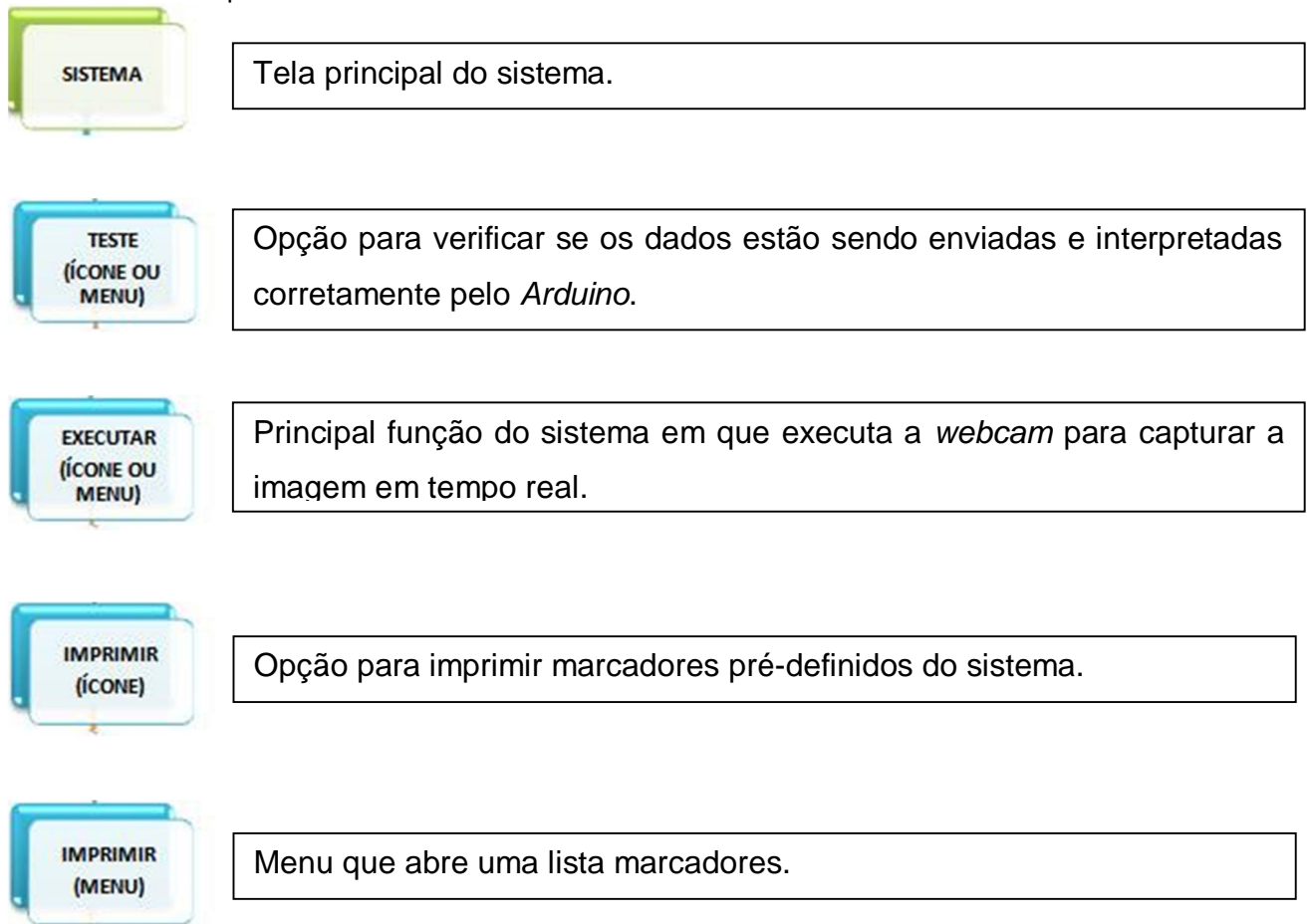
Dispositivo que captura a imagem e exibe na tela do sistema.

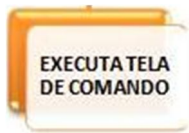


Na Figura 22 podemos observar a hierarquia estrutural de funcionamento e utilização do sistema.



Figura 22 - Hierarquia estrutural de funcionamento e utilização do sistema.
 Fonte: Elaborado pelo autor.





Tela por onde se controla a unidade robótica pelo teclado. Os movimentos são representados pelas teclas em forma de setas, em que cada uma delas assim que pressionadas envia um comando para o *Arduino*.



É possível controlar a unidade robótica (*Arduino*) apenas pelos marcadores. Cada marcador representa um comando que é enviado para o *Arduino* e executa um dos 5 movimentos, sendo eles: para frente, para trás, para esquerda, para direita e parar.



Tela de aviso caso usuário clique na opção "Imprimir" sem antes ter selecionado um marcador.



Tela com configurações da impressora e da impressão. Após clicar em "OK", a impressão inicia.



Lista com as 5 opções de marcadores que podem ser impressos.

Depois de instalados e configurados de acordo, começou o desenvolvimento do sistema, onde tem uma tela com três menus e quatro botões (Figura 23).

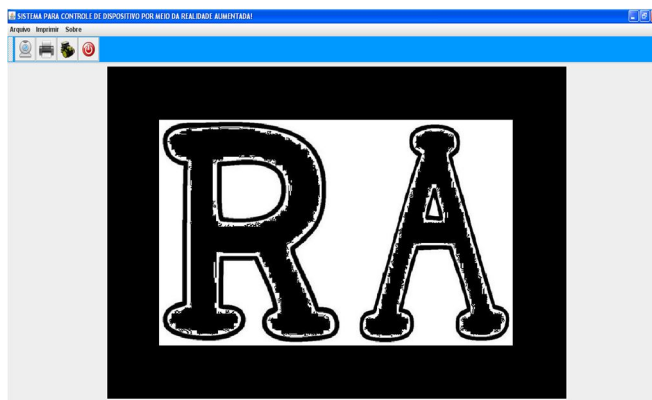


Figura 23 - Modelo do sistema.
Fonte: Elaborado pelo autor.

A principal função do sistema é no botão com uma imagem da *webcam* (Figura 24).



Figura 24 - Botão da *webcam*
Fonte: Elaborado pelo autor.

Ou também no menu “Arquivo” e sub menu “Executar” (Figura 25).



Figura 25 - Menu arquivo.
Fonte: Elaborado pelo autor.

Na Figura 26 mostra uma janela com a imagem capturada pela *webcam*.

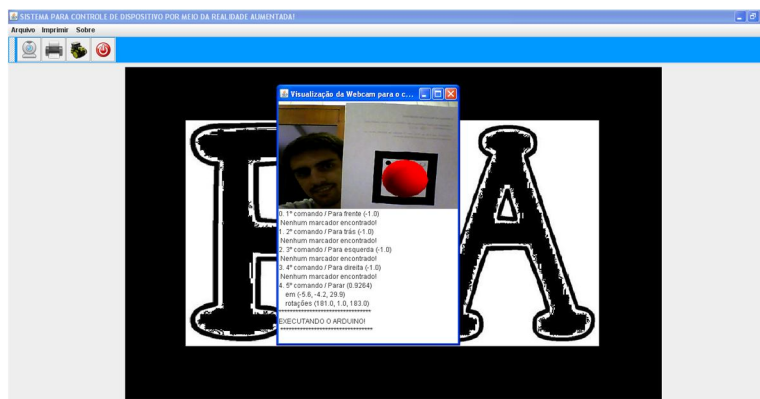


Figura 26 - Janela da *webcam*.
Fonte: Elaborado pelo autor.

No exemplo acima, o marcador foi identificado e criado a sobre posição de um objeto 3D, e também executado o comando para o *Arduino*.

A unidade robótica se trata de um protótipo de carrinho, onde o *Arduino* controla suas ações, como se fosse o cérebro humano, mas são necessários mais alguns componentes, como por exemplo: um servo (Figura 27) que controla o movimento de virar as rodas do carrinho, podendo chegar a uma angulação de 180°.



Figura 27 - Servos.
Fonte: Servo... (2011).

Um controle de velocidade, se necessário, conhecido como “Ponte HH” (Figura 28).

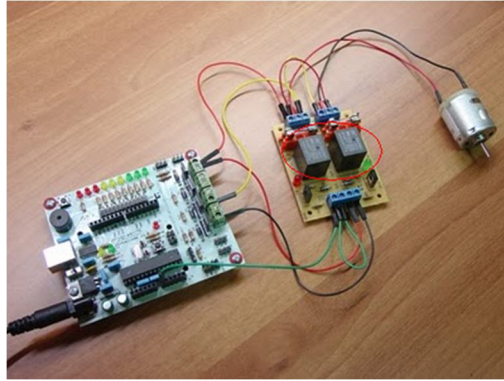


Figura 28 - Ponte HH.
Fonte: Lorenzo (2010).

E principalmente o motor, que pode ser um modelo *DC* (Figura 29).



Figura 29 - Motor *DC*.
Fonte: Maxon... (c2007).

É necessário também o revestimento, armação do carrinho ou a forma que achar conveniente.

O protótipo (Figura 30) de carrinho utilizado no projeto foi adquirido em forma de um *kit* contendo todos os componentes necessários para montá-lo, segue abaixo algumas fotos tiradas dele.

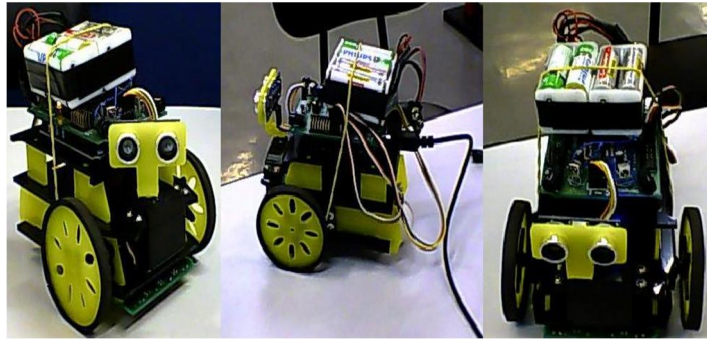


Figura 30 - Protótipo do carro.
Fonte: Elaborado pelo autor.

O uso das pilhas é indispensável, pois é ela que fornece energia ao *Arduino*.

Para a criação de um marcador, foi necessário o uso do *software ARToolKit Maker Generator*.

Mas antes foi necessário abrir o programa *paint* do *Windows*, ou utilizar um *software* com a mesma função, e desenhar um quadrado de cor preta com outro dentro, mas de cor branca, e um desenho da palma da mão com algumas partes bem pretas.

Foi necessário ser quadricular ou retangular, pois é assim que o *software* vai identificá-lo.

Em seguida, após instalado e posteriormente executado o *software ARToolKit Maker Generator*, executa uma janela com a imagem sendo capturada pela *webcam*, e nela foi mostrado o marcador criado e aguardado até ser reconhecido(Figura 31).

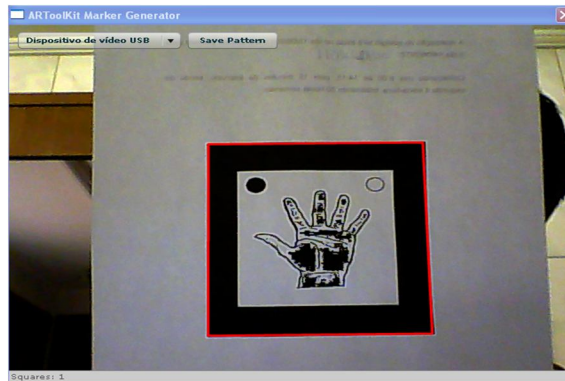


Figura 31 - *ARToolKit Maker Generator*
Fonte: Elaborado pelo autor.

Logo em seguida foi clicado em “*Save Pattern*”, digitado o nome de 5º comando e selecionado o diretório para salvar o arquivo.

Foi colocado o arquivo dentro da pasta “data” do projeto, que fica localizado na classe “TelaWebcam”, no parâmetro “5º comando”(Figura 32).

```
//Parar
ModeloMarcador mm5 = new ModeloMarcador("5º comando", "Parar.wrl", 0.12, false);
```

Figura 32 - Linha de código que carrega marcador.
Fonte: Elaborado pelo autor.

Quando se mostra o marcador no sistema, a biblioteca *NyARToolKit* fica procurando na imagem capturada pela *webcam* e se existe algum formato de quadrado que dentro dele contenha alguma informação igual ao da pasta “Data”.

É feita as comparações até que encontre uma correspondente.

Localizado o marcador, é sobreposto um objeto 3D no marcador e executado um comando que envia uma letra pela porta *USB*, mas que o *Arduino* a converte em Serial, ao *Arduino* (Figura 33).

```
arduino.enviaDados('p');
```

Figura 33 - Envia letra ao *Arduino*.
Fonte: Elaborado pelo autor.

Quando o *Arduino* recebe a letra ele verifica se tem alguma condição que confere com o dado recebido e então ser executado (Figura 34).

```

char tecla;
tecla = Serial.read();
if(tecla == 'f') {
  Servos[1].write(180) ;
  Servos[2].write(0) ;
  Servos[3].write(90) ;
  delay(15);}
if(tecla == 't') {
  Servos[1].write(0) ;
  Servos[2].write(180) ;
  Servos[3].write(90) ;
  delay(15); }
if(tecla == 'e') {
  Servos[1].write(0) ;
  Servos[2].write(0) ;
  Servos[3].write(180) ;
  delay(15); }
if(tecla == 'd') {
  Servos[1].write(180) ;
  Servos[2].write(180) ;
  Servos[3].write(0) ;
  delay(15); }
if(tecla == 'p') {
  Servos[1].write(70) ;
  Servos[2].write(70) ;
  delay(15); }
}

```

Figura 34 - Código de ativação dos movimentos.
Fonte: Elaborado pelo autor.

Reconhecida a comparação, o *Arduino* envia o comando.

Portanto dependendo de qual marcador for mostrado, ele executará um comando que sobrepõem um objeto 3D no marcador e executa um movimento do *Arduino* que está conectado via porta *USB*, mas que o converte em uma porta serial.

5 CONSIDERAÇÕES FINAIS

Com o desenvolvimento do sistema que foi proposto neste trabalho foi possível o controle da unidade robótica por meio da realidade aumentada e atingir os objetivos com êxito proporcionando novas áreas de estudo.

Contudo com o auxílio da realidade aumentada foi possível um controle com uma boa precisão da unidade robótica.

Portanto é possível a interação entre unidade robótica e realidade aumentada através do controle humano.

6 APRIMORIAMENTO FUTUROS

Webcam embutida na unidade robótica.

Comunicação entre o computador e o Arduino via *wireless*.

Controle do Arduino baseado em coordenadas que se encontra o marcador.

REFERÊNCIAS

- ARTEFATOS explosivos: *IEDs*. **Europa Newswire**, 2009. Disponível em: <<http://europanewsradio.typepad.com/europa-news-radio/2009/10/artefatos-explosivosieds.html>>. Acesso em: 08 jun. 2012.
- BREGA, J. R. F. et. al. O emprego de realidade aumentada na viabilização da Comunicação em libras. **Unesp**, 2008.p. 1-4. Disponível em: <<http://www2.fc.unesp.br/wrva/artigos/50457.pdf>>. Acesso em: 08 jun. 2012.
- BOEIRA, M. **Liberando porta paralela ou serial**, 2011. Disponível em: <<http://geekdevteam.blogspot.com.br/2011/04/liberando-porta-paralela-ou-serial.html>>. Acesso em: 08 jun. 2012.
- CARLOS, A. Instalando a *webcam* 300 da GOTEC. **Vivaolinux.com.br**, 2007. Disponível em: <<http://www.vivaolinux.com.br/dica/Instalando-a-webcam-300-da-GOTEC>>. Acesso em 08 jun. 2012.
- CAROLINA, L.. Dispositivo de Entrada. **Liscarol09.blogspot.com**, 2010. Disponível em: <<http://liscarol09.blogspot.com/2010/07/dispositivo-de-entrada.html>>. Acesso em: 08 jun. 2012.
- FONSECA, E. G. P.; BEPPU, M. M.. **Apostila Arduino**. Niterói, 2010. 22 f. Apostila.
- FILIPPO, D. et al. Ambientes Colaborativos de Realidade Virtual e Aumentada. In: KIRNER, C.; SISCOOTTO, R. (Ed.). **Realidade Virtual e Aumentada: conceitos, projeto e aplicações**. Porto Alegre: Editora SBC–Sociedade Brasileira de Computação, 2007.
- FURGERI, S.. **Java7: ensino didático**. São Paulo: Editora Érica Ltda, 2010.
- GASPARIN, M.. Indústria-automotiva.jpg. **Jornale.com.br**, 2011. Disponível em: <http://jornale.com.br/mirian/?attachment_id=14329>. Acesso em: 08 jun. 2012.

HEIMLICH, R.. *And The Point Of Safari For Windows is...* **Russellheimlich.com**, 2008. Disponível em: <<http://www.russellheimlich.com/blog/and-the-point-of-safari-for-windows-is/>>. Acesso em 08 jun. 2012.

ARNEIRO, D. et. al.. **interfacesdecomu.tripod.com**, 2006. Disponível em: <<http://interfacesdecomu.tripod.com/>>. Acesso em 08 jun. 2012.

INTRODUCCIÓN a la placa arduino. **arduino.cc/**, c2012. Disponível em: <<http://arduino.cc/es/Reference/Board?from=Guide.Board>>. Acesso em: 08 jun. 2012.

JORGE, E. **Artigo Java Magazine 57 – Aplicações multimídia em Java.** devmedia.com.br, c2012. Disponível em: <<http://www.devmedia.com.br/articles/viewcomp.asp?comp=9423>>. Acesso em: 08 jun. 2012.

KIRNER, C.; KIRNER, T.G. *Virtual Reality and Augmented Reality Applied to Simulation Visualization.* In: EL SHEIKH, A.A.R.; AL AJEELI, A.; ABU-TAIEH, E.M.O. (Ed.). **Simulation and Modeling: current technologies and applications.** Hershey-NY: IGI Publishing, 2008. p. 391-419. Disponível em: <<http://www.igi-global.com/Bookstore/Chapter.aspx?TitleId=28994>>. Acesso em: 08 jun. 2012.

KIRNER, C.; SISCOOTTO, R. **Realidade Virtual e Aumentada: conceitos, projeto, e aplicações.** Porto Alegre: Editora SBC – Sociedade Brasileira de Computação, 2007. Cap.1.

KIRNER, C.; TORI, R. Introdução à Realidade Virtual, Realidade Misturada e Hiper-realidade. In: KIRNER, C; TORI, R. (Ed.). **Realidade Virtual: conceitos e tendências.** São Paulo: Mania de livro, 2004.

LORENO, J. L. S.. **blog.eletronlivre.com.br**, 2010. Disponível em: <<http://blog.eletronlivre.com.br/2010/04/ponte-h-de-reles-com-program-me-e-placa.html>>. Acesso em: 08 jun. 2012.

MAGALHÃES, R.. Estudos tecnológicos (c/ exemplo). **Ricardofilipe**, 2009. Disponível em: <<http://ricardofilipe.blogs.ua.sapo.pt/1434.html>>. Acesso em: 08 jun. 2012.

MANO, R. Tipos de Dispositivos. **Users.rdc.puc-rio.br**, 1998. Disponível em: <<http://www.users.rdc.puc-rio.br/rmano/interfac.html>>. Acesso em: 08 jun. 2012.
MAXON motores DC. **treffer.com.br**, c2007. Disponível em: <<http://www.treffer.com.br/produtos/maxon/motoresdc.html>>. Acesso em: 08 jun. 2012.

MORIMOTO, C. E. *Java*. **Hardware.com.br**, 2005. Disponível em: <<http://www.hardware.com.br/termos/Java>>. Acesso em: 08 jun. 2012.

MORIMOTO, C. E. *Hardware Manual Completo*. **Hardware.com.br**, 2002. Disponível em: <<http://www.hardware.com.br/livros/hardware-manual/usb.html>>. Acesso em: 08 jun. 2012.

MORIMOTO, C. E. *Serial X Paralela X USB X Fireware: a evolução das portas*. **Hardware.com.br**, 2007. Disponível em: <<http://www.hardware.com.br/artigos/evolucao-portas/>>. Acesso em: 08 jun. 2012.

MORIMOTO, C. E. **Hardware o guia definitivo**. [S. l.]: Editora GDH Press e Sul Editores, 2007.

NORMANDIA, R. Entendendo a fundo *Qr code*. **Rodrigonormandia.com.br**, 2010. Disponível em: <<http://rodrigonormandia.com.br/2010/11/entendendo-a-fundo-qr-code/>>. Acesso em: 08 jun. 2012.

O QUE É *webcam*. **Guia Brasil blog**, 2009. Disponível em: <<http://www.guiabrasilblog.com/o-que-e-webcam/>>. Acesso em: 08 jun. 2012.

PIMENTA, R.; FARIA, R. **Porta Paralela e Porta de Série**. 2003.

PIRES, J. N. **Robótica: Das Máquinas Gregas à Moderna Robótica Industrial**. 2002.

RAMIRES, M.. Realidade aumentada: entenda a tecnologia que altera sua realidade. **Techtudo.com.br**, 2011. Disponível em: <<http://www.techtudo.com.br/artigos/noticia/2011/01/realidade-aumentada-entenda-tecnologia-que-altera-sua-realidade.html>>. Acesso em: 08 jun. 2012.

REDAÇÃO oficina da net. O que é o *NetBeans*? **Redação Oficina da Net**, 2008.
Disponível em: <http://www.oficinadanet.com.br/artigo/1061/o_que_e_o_NetBeans>.
Acesso em: 08 jun. 2012.

RIBEIRO, M. A. **Automação Industrial**. 4 ed. Salvador: Tek treinamento & Consultoria Ltda, 2001.

RODRIGUES, T. Dicas para Comprar uma *Webcam*. **Mundomax.com.br**, 2010. Disponível em: <<http://www.mundomax.com.br/blog/tag/o-que-e-webcam/>>. Acesso em: 08 jun. 2012.

SERVO motors. *Acroname Servos: Theory, Operation, and Comparison*. **acroname.com**. 2011. Disponível em: <<http://www.acroname.com/robotics/info/articles/servo/servo.html>>. Acesso em: 08 jun. 2012.

SILVA, G. A. Afinal quem é esse tal de *arduino*? bateriavirtual2010.blogspot.com.br, 2010. Disponível em: <<http://bateriavirtual2010.blogspot.com.br/2010/11/afinal-quem-e-esse-tal-de-arduino.html>>. Acesso em 08 jun. 2012.

TORRES, G.; LIMA, C. Construindo Protótipos Usando a Porta Paralela. **Clubedohardware.com.br**, 2005. Disponível em: <<http://www.clubedohardware.com.br/printpage/Construindo-Prototipos-Usando-a-Porta-Paralela/1147>>. Acesso em 08 jun. 2012.

TORI, R.; KIRNER, C.; SISCOOTTO, R. **Fundamentos e tecnologia de realidade virtual e aumentada**. Porto Alegre: Editora SBC – Sociedade Brasileira de Computação, 2006.

ZORZAL, E. R.; BUCCIOLI, A. A. B.; KIRNER, C. O uso da Realidade Aumentada no Aprendizado Musical. **Realidade aumentada**, 2005. Disponível em: <http://www.realidadeaumentada.com.br/artigos/WARV_Music.pdf>. Acesso em 08 jun. 2012.

ZORZAL, E.R.; KIRNER, C. Jogos Educacionais em Ambiente de Realidade Aumentada. **Realidade aumentada**, 2005. Disponível em: <http://www.realidadeaumentada.com.br/artigos/WRA2005_ZORZAL-KIRNER.pdf>. Acesso em 08 jun. 2012.

APÊNDICE A - MANUAL DO SISTEMA

1 INTRODUÇÃO AO SISTEMA

Este sistema consiste em um projeto desenvolvido por Rafael Moraes Canarin da Universidade do Sagrado Coração – USC, Bauru/SP, para a obtenção do título de graduação em Ciências da Computação no ano de 2011.

Foi criado para auxiliar no controle de unidades robóticas resultando em um maior desempenho, agilidade e facilidade na execução.

Com isso, contribuindo em operações de alto risco para o ser humano, como o desarmamento de uma bomba.

Este manual foi elaborado para auxiliar na utilização do sistema.

O sistema já vem configurado e que apenas precisa que o arquivo “rxtxSerial.dll” encontrado dentro da pasta “rxtx” seja colocado na pasta system32 do Windows XP, normalmente encontra-se neste caminho: “C:\WINDOWS\system32”.

É necessário também a instalação de dois softwares, Java3D e Java Media Frame (JMF), que se encontram dentro da pasta “Realidade Arduino”.

O sistema é 100% funcional sobre o Windows XP, mas em versões superiores sofre certas deficiências, como no Windows Seven.

2 MENU PRINCIPAL

A Figura 1 a seguir mostra a tela inicial após execução do sistema.

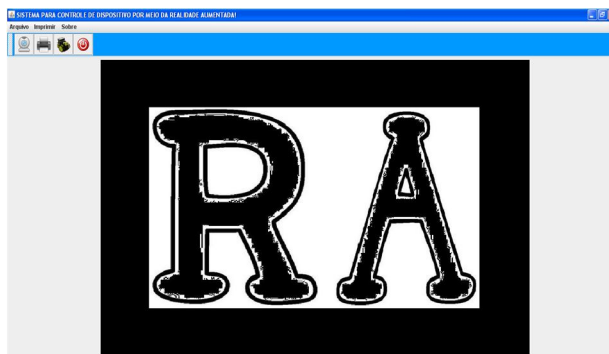


Figura 1 – Tela Principal.

A tela principal possui três menus (Figura 2).

Arquivo Imprimir Sobre

Figura 2 – Menus.

E quatro botões abaixo (Figura 3).

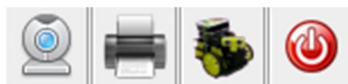


Figura 3 – Botões.

2.1 MENU ARQUIVO

Ao clicar no menu “Arquivo”, apresenta a seguinte imagem (Figura 4).



Figura 4 – Menu arquivo.

Três sub-menus são listados, sendo eles: “Executar”, “Testar Arduino” e “Sair”.

Ao clicar no “Executar” da lista ou no botão onde tem uma imagem de uma webcam (Figura 5).



Figura 5 – Botão da webcam.

É executado uma nova janela (Figura 6).

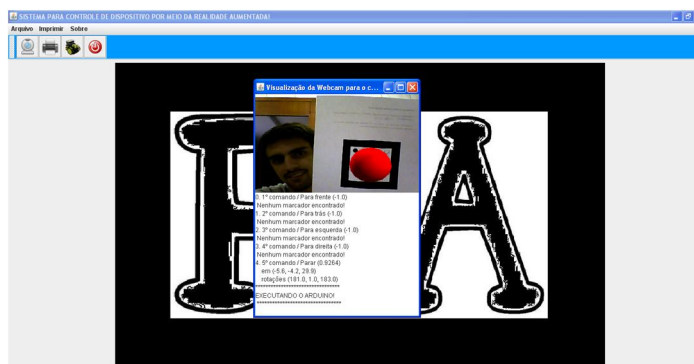


Figura 6 – Visualização da Webcam.

Nesta janela contém a imagem que está sendo capturada pela webcam. Dentro desta janela possui um quadrado branco, onde contém informações das posições e

rotações que se encontra um marcador quando identificado, como na imagem (Figura 6), será somente executado o comando para o arduino aquele que for identificado e gerado um objeto 3D.

Ao clicar no quadrado vermelho com um “X” branco, é emitida a seguinte mensagem (Figura 7).

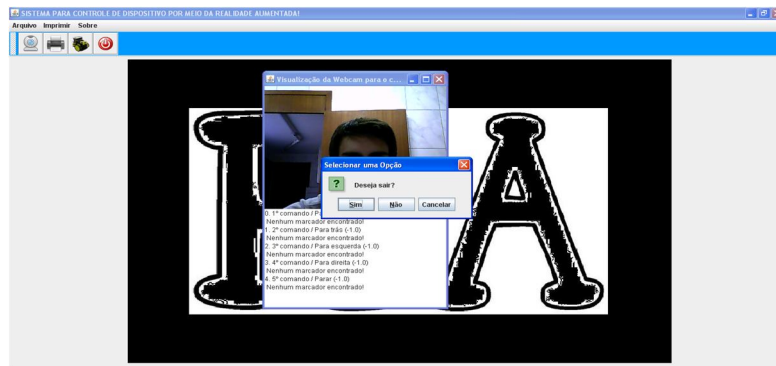


Figura 7 – Tela de confirmação.

Esta mensagem, “Deseja sair?”, requer a confirmação para fechar a tela, e para isso é preciso clicar na opção “Sim”, caso contrário não finaliza.

Ao clicar em “arquivo” e posteriormente em “Testar Arduino” ou clicar no terceiro botão (Figura 8), da esquerda para direita.



Figura 8 – Botão para testar Arduino.

É aberta outra janela (Figura 9).

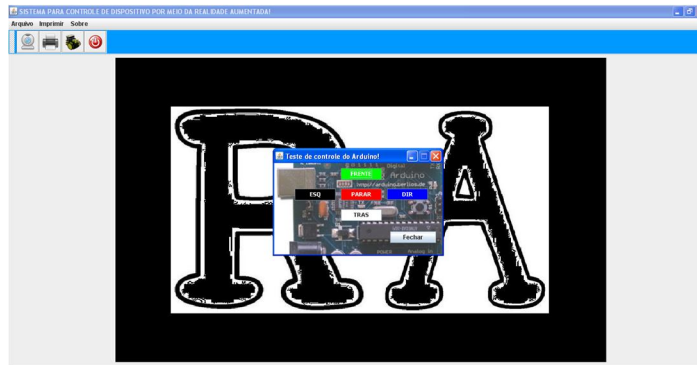


Figura 9 – Teste de controle do Arduino.

Nesta janela possui 5 botões coloridos com os respectivos nomes, “FRENTE” (verde), “TRÁS” (branco), “ESQ”(preto), “DIR”(azul), e “PARAR”(vermelho).

Ao clicar no botão “verde” ou aperta a tecla com a “seta pra cima”, o arduino se movimenta para frente.

Já se clicar no botão “branco” ou apertar a tecla com a “seta para baixo”, o arduino executa a “ré”.

Se clicar no botão “preto” ou apertar a tecla com a “seta para à esquerda”, ele vira para esquerda.

Se clicar no botão “azul” ou apertar a tecla de “seta para à direita”, o arduino vira para direita.

E caso clicar no botão “vermelho” ou na tecla “end”, o arduino fica parado.

Se a tela já estiver aberta e for clicado para abrir ela novamente, tanto pelo sub-menu ou pelo botão, uma mensagem (Figura 10) é emitida.



Figura 10 – Mensagem avisando de janela aberta

Só é possível sair dessa tela clicando no botão “Fechar”, pois no quadrado vermelho com um “X” branco não tem efeito.

Clicando em “arquivo” e depois em “sair”, apenas finaliza o sistema.

2.2 MENU IMPRIMIR

Caso clique primeiro no botão (Figura 11) com uma imagem de impressora, segundo da esquerda para direita, antes de clicar em “Imprimir” (Figura 12) é mostrado uma mensagem (Figura 13).



Figura 11 – Botão para impressão de marcador.



Figura 12 – Seleção de marcador para impressão.



Figura 13 – Mensagem ao clicar no botão.

Agora, se selecionar o marcador para impressão (Figura 14) e posteriormente clicar no botão de imprimir (Figura 11), é mostrado a seguinte tela (Figura 15).



Figura 14 – Marcador para imprimir selecionado.



Figura 15 – Tela de configuração para imprimir.

Nesta tela contém opções para a sua impressão, onde no primeiro campo (Nome), é o nome da impressora que vai utilizar.

Abaixo tem alguns detalhes sobre a sua impressora.

No quadrado pequeno (Imprimir em arq.), se ticado, é impresso como arquivo.

Na seção “Intervalo de impressão”, possui três opções: “Tudo”, imprimir todas as páginas, já na opção “Páginas de” é possível colocar um intervalo das páginas a serem impressas, e “Seleção”, onde seleciona apenas uma página.

E na seção “Cópias”, seleciona-se a quantidade de cópias do mesmo arquivo que é impresso.

2.3 MENU SOBRE

Ao clicar no menu “Sobre”(Figura 16).



Figura 16 – Tela ao clicar no menu “Sobre”.

E posteriormente clicar no sub-menu “Informações”, será aberta uma outra janela (Figura 17).

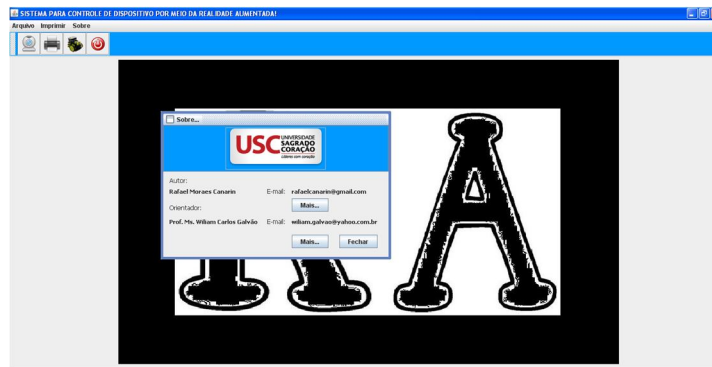


Figura 17 – Tela ao clicar no sub-menu “Informações”.

Nesta tela contém o logotipo da Universidade do Sagrado coração, e ao clicar sobre ela se abre uma nova janela (Figura 18).

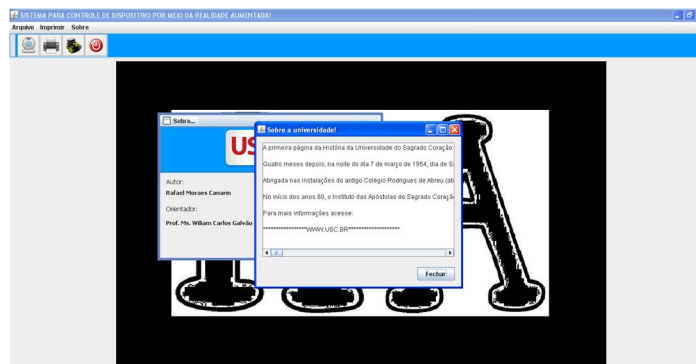


Figura 18 – Tela ao clicar sobre o logotipo da USC.

Na janela tem informações sobre a universidade e um botão “Fechar”, e caso seja clicado, a janela fecha.

Na janela anterior (Figura 17), tem mais dois botões e que Ambos chamam “Mais...”, porém com funções diferentes.

Caso seja clicado no botão da seção do “Autor”, se abre uma nova janela (Figura 19) com dados do autor do projeto.

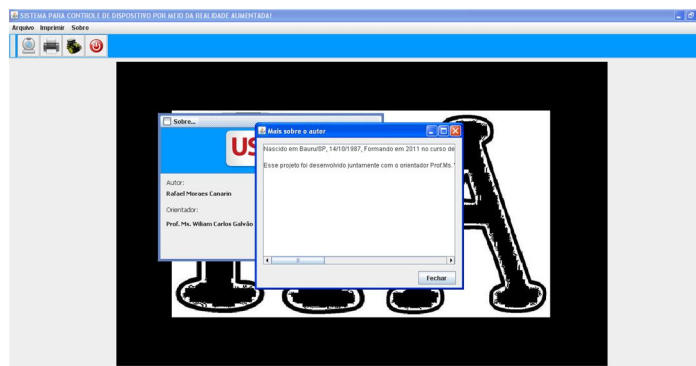


Figura 19 – Tela com informações do autor.

Agora, caso seja clicado no botão da seção do “Orientador”, se abre outra tela (Figura 20) com informações do orientador do projeto.

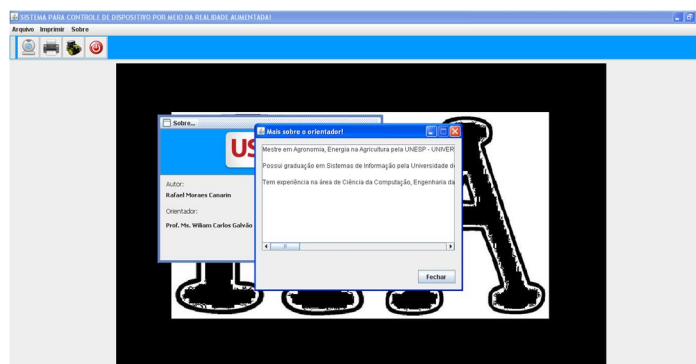


Figura 20 - Tela com informações do Orientador.

Se por ventura uma das janelas estiver aberta e for clicado novamente para abrir uma mesma, surgiu uma nova mensagem (Figura 21).

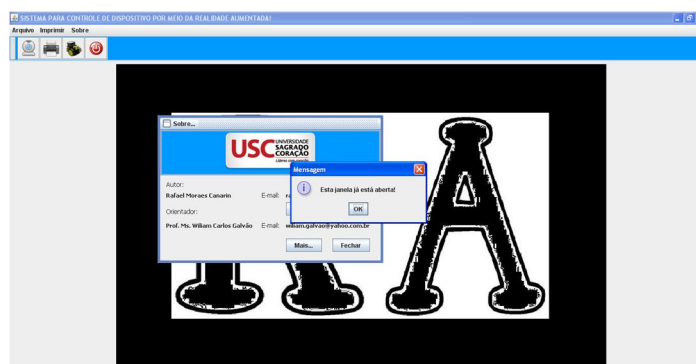


Figura 21 – Mensagem ao tentar abrir duas vezes a mesma tela.

3 CUIDADOS

Como todos os softwares ou sistemas, possui suas devidas limitações, para entender sobre o que pode estar ocorrendo, leia atentamente à próxima seção deste manual.

3.1 MARCADORES

Os marcadores do sistema se encontram no diretório “Realidade Arduino\src\Imagens” no formato “.jpeg”

É necessário ter no mínimo um marcador impresso para controlar o arduino pela Realidade Aumentada.

Pode ser impresso diretamente pela imagem que se encontra na pasta “Imagens” citada acima ou seguir os procedimentos no sistema como foi mostrado anteriormente.

Se caso o marcador que for utilizado esteja dobrado ou obstruído (qualquer coisa que ficar sobre ele), quando for exibi-lo na webcam, não será possível o reconhecimento do marcador e conseqüentemente não mostrará a sobreposição do objeto 3D e principalmente o movimento do arduino.

3.2 CÂMERA

Dependendo da qualidade da câmera será necessário que o usuário mantenha o marcador o mais próximo possível da câmera. Pois quanto menor a qualidade maior o desfoque da imagem capturada.

Caso (quando a câmera for de qualidade baixa) o usuário deverá exibir o marcador, o mais próximo possível da câmera/webcam, permitindo que a mesma consiga uma visão adequada do marcador, e realizar movimentos lentos ao movimentar o marcador, pois com baixa qualidade, a imagem pode distorcer e não ser reconhecida.

3.3 ILUMINAÇÃO

A iluminação pode gerar problemas ao usuário, por este motivo também deve ser levada em consideração.

A iluminação muito baixa não permite a câmera de localizar o marcador.

A iluminação excessiva também gera problemas, pois quando há iluminação excessiva, a iluminação gera um reflexo no marcador alterando sua forma original, fazendo com que a câmera não o reconheça como o marcador.

APÊNDICE B - ARTIGO**Sistema para controle de unidade robótica por meio da realidade aumentada**

Elvio Gilberto da Silva¹, Henrique Pachioni Martins¹, Rafael M. Canarin¹, Wiliam C. Galvão¹

**¹Centro de Ciências Exatas e Sociais Aplicadas – Universidade Sagrado Coração (USC)
Bauru, SP – Brasil**

***Abstract.** With new areas of work and highly dangerous situations, there is a need that man has more precision and control at it, and with it came the automation to replace tasks of human. Considering this and the absence of related projects that work aimed at developing a system in the Java programming language to control a robotic unit through Augmented Reality, may increase the ease and precision of each movement of the robot through the use of NyARToolKit library. The use of a prototype electronic open-source, in this case the Arduino, which is nothing more than a hardware that allows the sending of command by java language that allowed the interaction of computer-controlled engines. Arduino addition, it was necessary other components that accompany with the kit, such as two motors motion (control wheel), the head movement motor, the battery bank of 9.6 v. The system has three menus and submenus with some four buttons for easy interaction. A button captures the image of the webcam, and when one of the five markers are recognized, executes code for handling the Arduino, which can be in any direction except diagonally. Communication is done via USB (Universal Serial Bus), but that creates a virtual serial port, for transmitting the code to the Arduino. So any user can control or handle any type of robotic unit with a simple movement.*

Resumo. *Com as novas áreas de trabalho e situações de alta periculosidade, surge há necessidade em que o homem tenha mais precisão e controle no que faz, e com isso a automação veio para substituir as tarefas do ser humano. Tendo em vista isso e a ausência de projetos correlacionados esse trabalho teve como objetivo desenvolver um sistema na linguagem de programação Java para controlar uma unidade robótica por meio da Realidade Aumentada, podendo aumentar a facilidade e precisão de cada movimento do robô, por meio da utilização da biblioteca NyARToolKit. A utilização de um protótipo eletrônico open-source (código aberto), neste caso o Arduino, que nada mais é que um hardware que permite o envio de comando por linguagem Java que possibilitou a interação do computador com controle de motores. Além do Arduino, foram necessários outros componentes que acompanham junto com o kit, como dois motores de movimento (controle das rodas), motor do movimento da cabeça, banco de baterias de 9,6V. O sistema possui três menus com alguns submenus e quatro botões para facilitar a interação. Um dos botões captura a imagem da webcam, e quando um dos cinco marcadores forem reconhecidos, executa um código para a movimentação do Arduino, que pode ser para qualquer direção exceto diagonal. A comunicação é feita pela porta USB (Universal Serial Bus), mas que é criada uma porta serial virtual, para a transmissão dos códigos ao Arduino. Podendo assim qualquer usuário controlar ou manusear qualquer tipo de unidade robótica com um simples movimento.*

1 Introdução

A cada dia surgem novas pesquisas sobre como melhorar e facilitar as operações que exijam maior grau de eficiência e precisão nos procedimentos. Tal questão nota-se nas máquinas das indústrias, montadoras, metalúrgicas e robôs que desenvolvem tarefas com funções similares a de um ser humano. Assim como nos locais em que a acessibilidade para humanos são quase impossíveis ou de alto risco.

Diante deste contexto, ressalta-se a necessidade em adotar-se automação com o uso de dispositivos (hardwares) e os softwares para a realização de determinadas tarefas.

Embora as duas coisas sejam igualmente importantes, existe uma distinção entre o "hardware", que inclui todos os componentes físicos, como o processador, memória, placa-mãe, etc. e o "software", que inclui o sistema operacional, os programas e todas as informações armazenadas. (MORIMOTO, 2007, p.30).

“É o enriquecimento do ambiente real com objetos virtuais, usando algum dispositivo tecnológico, funcionando em tempo real.” (KIRNER; SISCOUTTO, 2007, p. 10).

“Realidade Aumentada, quando a parte predominante é o mundo real e, Virtualidade Aumentada quando ocorre o contrário, ou seja, existe a predominância no ambiente virtual.” (KIRNER; TORI, 2004, p. 15)

“A Realidade Virtual (RV) é uma “interface avançada do usuário” para acessar aplicações executadas no computador, propiciando a visualização, movimentação e interação do usuário, em tempo real, em ambientes tridimensionais gerados por computador.” (KIRNER; SISCOUTTO, 2007, p. 7).

“A Realidade Misturada pode receber duas denominações: Realidade Aumentada, quando o ambiente principal ou predominante é o mundo real e, Virtualidade Aumentada, quando o ambiente principal ou predominante é o mundo virtual”. (KIRNER; TORI, 2004, p. 15).

O *NyARToolKit* é uma ferramenta criada para atender aos programadores em *Java* para aplicação na Realidade Aumentada.

Com a utilização da ferramenta é possível criar aplicações com as imagens capturadas pela *webcam* de forma simplificada e eficaz, oferecendo muitas variedades para o desenvolvimento de aplicações de Realidade Aumentada e também a Realidade Virtual.

O *Java* é uma linguagem de programação de alto nível, orientada a objetos e multiplataforma.

“A programação orientada a objetos é hoje universalmente adotada como padrão de mercado. Muitas linguagens tradicionais foram aperfeiçoadas para implementar essa nova forma de trabalho, e *Java* já nasceu assim.”(FURGERI, 2010, p. 18).

“O *Java* é uma linguagem de programação multiplataforma, com uma sintaxe até certo ponto parecida com o *C++*, porém com bibliotecas diferentes. Os programas em *Java* podem ser executados em qualquer sistema operacional, desde que o interpretador esteja instalado.” (MORIMOTO, 2005).

O Arduino (Figura 1) é uma plataforma de prototipagem eletrônica open-source que se baseia em *hardware* e *software* flexível de fácil uso.

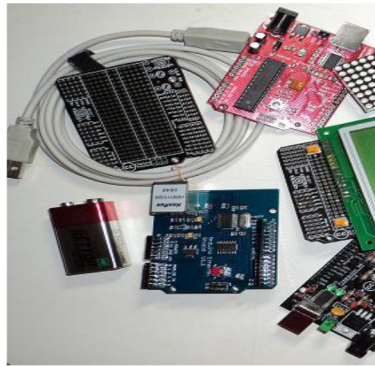


Figura 1 - Exemplos de Arduino.
Fonte: Elaborado pelo autor.

Arduino é uma plataforma de computação física (são sistemas digitais ligados a sensores e atuadores, que permitem construir sistemas que percebam a realidade e respondem com ações físicas), baseada em uma simples placa de Entrada/Saída microcontrolada e desenvolvida sobre uma biblioteca que simplifica a escrita da programação em C/C++.(FONSECA,2010, p. 2)

É um kit de desenvolvimento capaz de interpretar variáveis no ambiente e transformá-las em sinal elétrico correspondente, através de sensores ligados aos seus terminais de entrada, e atuar no controle ou acionamento de algum outro elemento eletro-eletrônico conectado ao terminal de saída. (FONSECA, 2010, p. 2)

Desenvolver um sistema em *Java* para controlar uma unidade robótica por meio da Realidade Aumentada.

Fazer uma ampla revisão da bibliografia para contextualização do assunto.

Realizar um estudo específico da linguagem de programação *Java* para sua aplicação na construção do sistema para o controle da unidade robótica.

Implementar a tecnologia de Realidade Aumentada no controle da unidade robótica.

2 Revisão Bibliográfica

Dependendo desta forma de combinação pode-se chegar as vertentes da Realidade Misturada, que vai desde um ambiente totalmente real, passando pela Realidade Aumentada (onde há a inserção de objetos virtuais dentro do ambiente real), pela Virtualidade Aumentada (onde há a inserção de objetos reais em ambientes virtuais), até se chegar a um ambiente totalmente virtual. (BREGA et al., 2008, p. 1)

“[...] “hiper-realidade é a capacidade tecnológica de combinar realidade virtual, realidade física, inteligência artificial e inteligência humana, integrando-as de forma natural para acesso do usuário”.” (KIRNER; SISCOOTTO, 2007, p. 12).

A *IDE NetBeans* é um ambiente de desenvolvimento multiplataforma, uma ferramenta que auxilia programadores a escrever, compilar, debugar e instalar aplicações, foi arquitetada em forma de uma estrutura reutilizável que visa simplificar o desenvolvimento e aumentar a produtividade pois reúne em uma única aplicação todas estas funcionalidades. Totalmente escrita em *Java*, mas que pode suportar qualquer outra linguagem de programação ou linguagem que desenvolva com *Swing*, algumas das linguagens que o *NetBeans* suporta são o *C*, *C++*, *Ruby*, *PHP*, *XML* e linguagens *HTML*. (REDAÇÃO..., 2008).

A qualidade de imagem é outro fator. A qualidade de imagem de uma *webcam* é o resultado da taxa de quadros e da resolução da câmera, embora a qualidade e a capacidade das lentes também tenham um papel importante. A taxa de quadros é expressa em *FPS* (quadros por segundo), com taxas de quadros comuns para *webcams* em 15 *FPS* e 30 *FPS*. (RODRIGUES, 2010).

Interfaces de comunicação são padrões lógicos e físicos em relação a como são ligados e transmitidos os sinais entre equipamentos e meios de comunicação. Fisicamente constituem de conectores que têm uma pinagem toda específica onde são transmitidas as informações e também são emitidos dados de controle, tais como início de uma transmissão, finalização, confirmação de recebimento, etc. O tipo de interface geralmente está vinculado à velocidade da rede de comunicação. (ARNEIRO, 2006)

“O *USB* é um novo padrão para a conexão de periféricos externos. Suas principais armas são a facilidade de uso e a possibilidade de se conectar vários periféricos a uma única porta *USB*.” (MORIMOTO, 2002)

Com exceção talvez do *PCMCIA*, o *USB* é o primeiro barramento para micros PC realmente Plug-and-Play. Podemos conectar periféricos mesmo com o micro ligado, bastando fornecer o driver do dispositivo para que tudo funcione sem ser necessário nem mesmo reinicializar o micro. A controladora *USB* também é suficientemente inteligente para perceber a desconexão de um periférico. (MORIMOTO, 2002)

3 Metodologia

Inicialmente foram instalados os *softwares*, como o *NetBeans* na versão 7.1 para programação em *Java*, *ARToolKit Maker Generator* para criação dos marcadores, *Java3D* para os objetos 3D, *Java Media Frame* para a captura da *webcam* e as bibliotecas *NyARToolKit* para o desenvolvimento da Realidade Aumentada e o *RXTX* para a comunicação com a porta Serial.

Após isso será necessário reiniciar o computador, para efetivação da configuração.

Depois de instalados e configurados de acordo, começou o desenvolvimento do sistema, onde tem uma tela com três menus e quatro botões (Figura 2).



Figura 2- Modelo do sistema.
Fonte: Elaborado pelo autor.

A principal função do sistema é no botão com uma imagem da *webcam* (Figura 3), ou no menu “Arquivo” e sub menu “Executar”(Figura 4), onde abre uma janela com a imagem capturada pela *webcam* (Figura 5).



Figura 3- Botão da *webcam*
Fonte: Elaborado pelo autor.

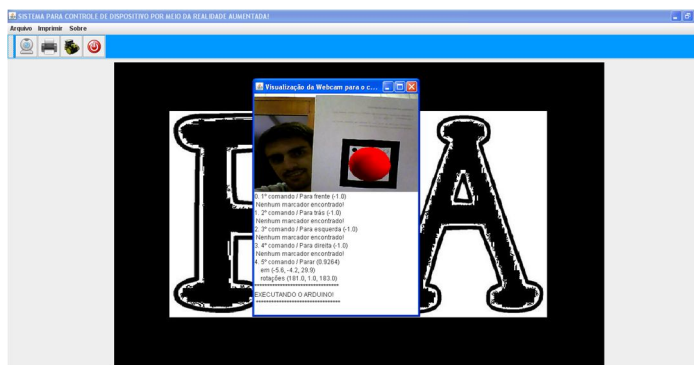


Figura 4 – Visualização da Webcam.
Fonte: Adaptado pelo autor.

Nesta janela contém a imagem que está sendo capturada pela webcam. Dentro desta janela possui um quadrado branco, onde contém informações das posições e rotações que se encontra

um marcador quando identificado, e somente executará o comando para o Arduino aquele que for identificado e gerado um objeto 3D.

O protótipo (Figura 5) de carrinho utilizado no projeto foi adquirido em forma de um *kit* contendo todos os componentes necessários para montá-lo, segue abaixo algumas fotos dele.

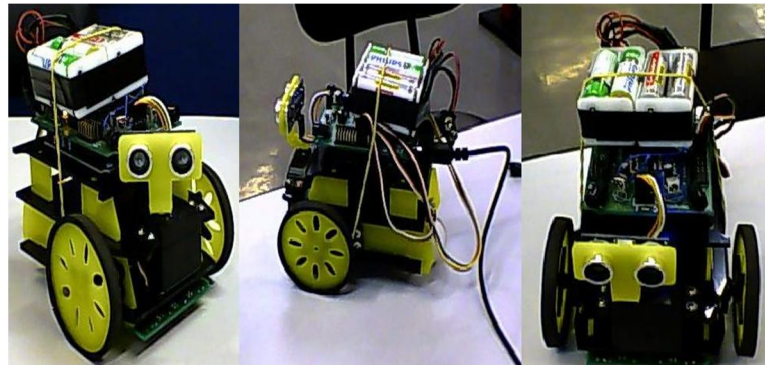


Figura 5 - Protótipo do carro
Fonte: Elaborado pelo autor.

4 Considerações Finais

Com o desenvolvimento do sistema que foi proposto neste trabalho foi possível o controle da unidade robótica por meio da realidade aumentada e atingir os objetivos com êxito proporcionando novas áreas de estudo.

Contudo com o auxílio da realidade aumentada foi possível um controle com uma boa precisão da unidade robótica.

Portanto é possível a interação entre unidade robótica e realidade aumentada através do controle humano.

Referências

- BREGA, J. R.F. et al. O emprego de realidade aumentada na viabilização da Comunicação em libras. **Unesp**, 2008.p. 1-4. Disponível em: <<http://www2.fc.unesp.br/wrva/artigos/50457.pdf>>. Acesso em: 08 jun. 2012.
- FONSECA, E. G. P.; BEPPU, M. M.. **Apostila Arduino**. Niterói, 2010. 22 f. Apostila.
- FURGERI, S.. **Java7: ensino didático**. São Paulo: Editora Érica Ltda, 2010. 320p.
- ARNEIRO, D. et. al.. **interfacesdecomu.tripod.com**, 2006. Disponível em: <<http://interfacesdecomu.tripod.com/>>. Acesso em: 08 jun. 2012.
- KIRNER, C.; SISCOOTTO, R. **Realidade Virtual e Aumentada: conceitos, projeto, e aplicações**. Porto Alegre: Editora SBC – Sociedade Brasileira de Computação, 2007. Cap.1, p. 7-19.
- KIRNER, C.; TORI, R. Introdução à Realidade Virtual, Realidade Misturada e Hiper-realidade. In: KIRNER, C; TORI, R. (Ed.). **Realidade Virtual: conceitos e tendências**. São Paulo: Mania de livro, 2004. p. 15.
- MORIMOTO, C. E..**Hardware Manual Completo. Hardware.com.br**, 2002. Disponível em: <<http://www.hardware.com.br/livros/hardware-manual/usb.html>>. Acesso em: 08 jun. 2012.
- MORIMOTO, C. E. **Hardware o guia definitivo**. [S. l.]: Editora GDH Press e Sul Editores, 2007.
- MORIMOTO, C. E. **Java. Hardware.com.br**, 2005. Disponível em: <<http://www.hardware.com.br/termos/Java>>. Acesso em: 08 jun. 2012.
- REDAÇÃO oficina da net. O que é o *NetBeans*? **Redação Oficina da Net**, 2008. Disponível em:<http://www.oficinadanet.com.br/artigo/1061/o_que_e_o_NetBeans>. Acesso em: 08 jun. 2012.
- RODRIGUES, T. Dicas para Comprar uma *Webcam*. **Mundomax.com.br**, 2010. Disponível em: <<http://www.mundomax.com.br/blog/tag/o-que-e-webcam/>>. Acesso em: 08 jun. 2012.