

**UNIVERSIDADE DO EXTREMO SUL CATARINENSE – UNESC  
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**BRUNO NUNES BITTENCOURT**

**ANIMAÇÃO POR CAPTURA DE MOVIMENTO COM O USO DO KINECT: UMA  
APLICAÇÃO PARA INSTRUÇÃO DE DANÇA**

**CRICIÚMA**

**2013**

**BRUNO NUNES BITTENCOURT**

**ANIMAÇÃO POR CAPTURA DE MOVIMENTO COM O USO DO KINECT: UMA  
APLICAÇÃO PARA INSTRUÇÃO DE DANÇA**

Trabalho de Conclusão de Curso, apresentado para a obtenção do grau de Bacharel no curso de Ciência da Computação da Universidade do Extremo Sul Catarinense.

Orientadora: Prof<sup>a</sup>. Msc. Leila Laís Gonçalves

**CRICIÚMA**

**2013**

**BRUNO NUNES BITTENCOURT**

**ANIMAÇÃO POR CAPTURA DE MOVIMENTO COM O USO DO KINECT: UMA  
APLICAÇÃO PARA INSTRUÇÃO DE DANÇA**


Trabalho de Conclusão de Curso aprovado  
pela Banca Examinadora para obtenção do  
Grau de Bacharel no curso de Ciência da  
Computação da Universidade do Extremo  
Sul Catarinense – UNESC, com Linha de  
Pesquisa em Computação Gráfica.

Criciúma, 28 de Novembro de 2013.

**BANCA EXAMINADORA**

  
Prof<sup>a</sup> Leila Lais Gonçalves – MSc – (UNESC) – Orientadora

  
Prof. Evânio Ramos Nicoleit – MEng – (UNESC)

  
Prof. Gustavo Bisognin – MSc – (UNESC)

## RESUMO

As aplicabilidades da captura de movimento vêm sendo mais exploradas pelas mais diversas áreas, sendo possível a sua utilização para auxiliar na ampliação de acessibilidade e qualidade de rotinas e exercícios físicos. Para uma melhor compreensão, as rotinas e movimentos de dança foram o foco deste experimento, a fim de demonstrar uma das diversas aplicações desta ferramenta. Para que isso ocorra é necessário um estudo sobre a modelagem do corpo humano, a criação de um modelo para compreensão do fluxo de informações e a execução de testes para determinar sua eficiência. Em posse destas informações, a criação de uma ferramenta ou protótipo se torna possível, sendo necessária, previamente, a identificação dos pré-requisitos, sendo que, paralelamente ao desenvolvimento da ferramenta, foram feitos testes para verificar se seu funcionamento está de acordo com o estipulado, além de mostrar progressivamente a eficiência de tal ferramenta, auxiliando na sua compreensão, aplicação e melhoria. Nos testes realizados após a conclusão do desenvolvimento do protótipo foi possível obter uma animação dos movimentos feitos pelo ator, além da identificação da abrangência de aplicação do algoritmo, principalmente na área da saúde, e suas possíveis melhorias, como o preenchimento da estrutura a fim de criar uma representação mais fiel e detalhada do corpo humano ou a simples acoplagem de músicas nos vídeos.

**Palavras-chave:** Captura de movimento. Kinect. Processing. PKinect. Animação computacional

## **ABSTRACT**

The applicability of motion capture are being explored by several areas, its use to assist in expanding the accessibility and quality of physical exercises and routines are possible. For a better understanding, routines and dance movements were the focus of this experiment in order to demonstrate the various applications of this feature. For this to occur a study on modeling the human body, creating a model for understanding information flow and execution of tests is required to determine its efficiency. In this information, the creation of a tool or prototype becomes possible, identification of the prerequisites is required in advance, and, alongside the development of the tool, tests were done to verify that its operation is in accordance with the stipulated, and gradually show the efficiency of this tool, aiding in their understanding, application and improvement. In tests after completion of the development of the prototype, was possible to obtain an animation of the movements made by the actor, in addition to identifying the scope of application of the algorithm, especially in the area of health, and its possible improvements , such as filling the structure in order to create a more faithful and detailed representation of the human body or the simple coupling of the music videos.

**Keywords:** Motion capture. Kinect. Processing. PKinect. Computer animation.

## LISTA DE ILUSTRAÇÕES

Figura 1 - Hierarquia do corpo humano.....	17
Figura 2 - Esferas e seus respectivos raios de influência.....	17
Figura 3 - Passos básicos de vaneirão .....	18
Figura 4 - Posição principais dos braços no Ballet .....	19
Figura 5 – Cena do filme <i>O Vingador do Futuro</i> .....	20
Figura 6 - Aplicação da técnica de <i>keyframe</i> .....	21
Figura 7 – Componentes do Microsoft Kinect Sensor.....	26
Figura 8 - Modelo para geração automática de animação através da captura de movimento através do <i>Kinect</i> .....	32
Figura 9 – Tela inicial do ambiente Processing .....	34
Figura 10 - Exibição 4 formas de captura utilizando <i>Kinect</i> e a biblioteca <i>PKinect</i> ...	35
Figura 11 - Declaração das variáveis globais da aplicação .....	36
Figura 12 - Desenho dos ossos do tronco e cabeça .....	37
Figura 13 – Captura obtida com a utilização do protótipo.....	39

## LISTA DE TABELAS

Tabela 1 - Limites rotacionais das articulações .....	16
Tabela 2 - Sistemas existentes para captura de movimento .....	23

## **LISTA DE ABREVIATURAS E SIGLAS**

Fps	Frames por segundo
JVM	Java Virtual Machine
Kinect	Microsoft Kinect Sensor
MIT	Massachusetts Institute of Technology
SCAM	Sistema de captura e análise de movimento
Sun	Sun Microsystems
VM	Virtual Machine

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	<b>9</b>
1.1 OBJETIVO GERAL .....	11
1.2 OBJETIVOS ESPECÍFICOS .....	11
1.3 JUSTIFICATIVA .....	11
1.4 ESTRUTURA DO TRABALHO.....	13
<b>2 DANÇA, CAPTURA DE MOVIMENTO E ANIMAÇÃO COMPUTACIONAL</b> .....	<b>15</b>
2.1 DANÇA .....	15
<b>2.1.1 Representação do corpo humano</b> .....	<b>15</b>
2.1.1.1 Personagem segmentado .....	16
2.1.1.2 Personagem criado com <i>metaballs</i> .....	17
<b>2.1.2 Formas de representação</b> .....	<b>18</b>
2.2 CAPTURA DE MOVIMENTO .....	19
2.3 ANIMAÇÃO COMPUTACIONAL .....	20
2.4 ANIMAÇÃO POR CAPTURA DE MOVIMENTO.....	21
<b>3 TECNOLOGIA PARA ANIMAÇÃO E CAPTURA DE MOVIMENTO</b> .....	<b>23</b>
3.1 CAPTURA DE MOVIMENTO .....	23
<b>3.1.1 Kinect</b> .....	<b>25</b>
3.2 DESENVOLVIMENTO .....	27
<b>4 GERADOR DE VÍDEO A PARTIR DA CAPTURA DE MOVIMENTO</b> .....	<b>30</b>
4.1 METODOLOGIA .....	30
<b>4.1.1 Levantamento bibliográfico</b> .....	<b>30</b>
<b>4.1.2 Análise das funcionalidades</b> .....	<b>31</b>
<b>4.1.3 Modelo de geração automática</b> .....	<b>32</b>
<b>4.1.4 Eclipse e Jnect</b> .....	<b>33</b>
<b>4.1.5 Processing e SimpleOpenNI</b> .....	<b>34</b>
<b>4.1.6 Processing e PKinect</b> .....	<b>35</b>
4.2 RESULTADOS OBTIDOS.....	38
4.3 APRESENTAÇÃO, DISCUSSÃO E ANÁLISE DOS RESULTADOS OBTIDOS .	40
<b>5 CONCLUSÃO</b> .....	<b>42</b>
<b>REFERENCIAS</b> .....	<b>43</b>
<b>REFERENCIAS COMPLEMENTARES</b> .....	<b>46</b>
<b>ANEXOS</b> .....	<b>49</b>
<b>APÊNDICE</b> .....	<b>54</b>

## 1 INTRODUÇÃO

A melhoria na qualidade de vida tem sido, nos últimos tempos, uma busca contínua levando as pessoas a mudanças na alimentação, rotina, lugares frequentados e adesão a algum tipo de atividade física (MOREIRA, 2013). As atividades físicas disponíveis são diversas e dentre as mais procuradas destacam-se a musculação, caminhadas, andar de bicicleta, natação, ginástica e dança (PREVIDÊNCIA, 2013; SALVE, 2013).

Na busca por atividade orientada por profissionais, as pessoas se deparam com diferentes problemas na adesão. Um desses problemas é o encaixe da atividade em sua rotina referente aos horários de oferta. Apesar da flexibilidade dos horários, já possibilitada por academias ou outros estabelecimentos, nem todas as modalidades de atividades estão disponíveis.

No caso da atividade de dança, muitos estabelecimentos dependem de número de alunos para fechar turmas. Outra situação encontrada se refere ao nível de aperfeiçoamento desejado (iniciante, intermediário ou avançado). Para atender as questões referentes ao horário, turmas e necessidade específica de especialização, dentre outras, foram criados cursos à distância em dança. Os cursos são elaborados de forma que o estudante possa utilizar seu tempo livre para aprender e exercitar seus conteúdos. Os recursos para disponibilização de conteúdos variam de material impresso a vídeo, aulas gravadas ou ao vivo. Cursos oferecidos pelo Estúdio de Dança Paulo Aguiar ([www.pauloaguiar.com.br](http://www.pauloaguiar.com.br)) e pela Escola Overdance ([www.overdance.net](http://www.overdance.net)) são exemplos da utilização de recursos de vídeo para a criação de vídeo-aulas de dança, ajudando, inclusive, na maior abrangência dos cursos devido às vendas *on-line*.

Apesar de ser um método de disponibilização mais utilizado, o vídeo apresenta deficiências na apresentação de passos de dança que precisam ser mostrados de forma mais lenta ou passo a passo, em virtude das limitações e especificidades próprias dos movimentos, articulação de determinada parte do corpo, do equilíbrio necessário para tal (EVANS, 2013, tradução nossa). Assim, a elaboração de um material didático para uma aula que demonstre de forma eficiente

cada movimento requer recursos no qual seja possível mostrar as minuciosidades do movimento identificando as partes do corpo a serem movimentadas e em velocidade reduzida. Considerando que o uso de paradas, truques de imagens ou efeitos de desaceleração podem prejudicar a qualidade da demonstração e reprodução do movimento como também a *performance* do instrutor na captação de imagens do movimento. Uma possível solução é o uso da animação computacional com base na captura de movimentos do instrutor durante a execução da dança gravando os passos do exercício.

Diferentes tecnologias estão disponíveis para a captura de movimento, técnica muito utilizada para filmes ou simulação de ações. Em função de seus altos custos, anteriormente essa tecnologia era utilizada somente por laboratórios ou estúdios de cinema. Porém, alguns recursos de tecnologias para captura de movimento estão disponíveis e mais acessíveis como é o caso do *Kinect*. O *Kinect* é um aparelho desenvolvido pela *Microsoft* inicialmente para o console *Xbox*. Ele funciona captando os movimentos do jogador e os comparando com os movimentos dados pelo jogo ou fazendo com que os personagens do jogo se movimentem com base nos gestos de seu jogador (QUEIROZ, 2013).

Considerando o uso do *Kinect* para a captura dos movimentos, de bibliotecas de um framework de desenvolvimento (por exemplo, *OpenKinect*, *OpenNI* ou *Microsoft Kinect SDK*) e de *plug-ins* para uso dos movimentos capturados (por exemplo, o *plug-in jnect*) fica ainda a lacuna da geração da animação, o mais automática possível sem a necessidade da intervenção direta de um programador e/ou animador.

Sendo assim, este trabalho visa desenvolver, a partir da técnica animação por captura de movimentos, um modelo para geração automática de animação de passos de dança com o uso do *Kinect* e de tecnologias para desenvolvimento como Java, *Eclipse IDE*, *Framework Microsoft Kinect SDK* e o *plug-in jnect*.

Com a geração da animação, pretende-se contribuir no aumento da qualidade do material didático de aulas de dança a distância, propiciando a

demonstração mais eficiente dos movimentos nos passos identificando as partes do corpo a serem movimentadas.

## 1.1 OBJETIVO GERAL

Desenvolver um modelo computacional para geração automática de animação de passos de dança a partir captura de movimentos com a utilização do *Kinect*.

## 1.2 OBJETIVOS ESPECÍFICOS

- Compreender a captura de movimentos descrevendo tecnologias e técnicas de animação computacional envolvidas;
- Representar e reconstituir movimentos de passos de dança por meio de animação computacional e;
- Desenvolver um protótipo para a geração automática de uma animação de passos e movimentos de dança, a partir da captura de movimentos com a utilização do *Kinect*.

## 1.3 JUSTIFICATIVA

Apesar de o nome parecer novidade, técnicas de captura de movimento já são amplamente utilizadas pelos estúdios de *Hollywood* para a criação e animação de personagens. Tais técnicas podem ser observadas em, praticamente, todos os filmes de animação da atualidade, proporcionando uma maior realidade no movimento e, se bem trabalhadas, chegam a fazer o personagem demonstrar emoções perfeitamente. Mas vale lembrar que tais estúdios utilizam equipamentos caríssimos e possuem uma equipe especializada para manuseá-los.

Com a chegada do *Kinect Sensor* tal técnica passou a ficar mais acessível à população devido a seu baixo custo, em comparação com outros equipamento que

utilizem o seu principal recurso, uma Câmera de Tempo de Voo (TOF – *time of flight*). Tal câmera já é conhecida desde 1998, porém seu preço sempre foi muito elevado, as melhores custavam aproximadamente US\$ 9.000, o que resultou no difícil acesso a elas pela maior parte da população (QUEIROZ, 2013). Sendo que o motivo de tal preço é o modo como funcionam. Resumidamente, elas trabalham medindo o tempo que luz leva para percorrer a distância de uma fonte de luz para o objeto e do objeto para a fonte (SCHMIDT; JÄHNE, 2013, tradução nossa), tornando possível a obtenção de um modelo 3D da imagem capturada. (CUI et al, 2013, tradução nossa).

Com tal modelo seria possível a criação ou modelagem de alguma ferramenta que possa vir a utilizá-lo para representar movimentos do corpo humano, utilizando uma série de fotos sequenciadas, ou mesmo, gerando um modelo esquelético do objeto e cobrindo cada parte com uma imagem representativa ou vetores delimitando-as.

Observando estes pontos, fica visível a possibilidade criação de uma imagem com uma boa qualidade. E, posteriormente, uma animação. Esta que poderá ser utilizada para os mais variados fins desde simples vídeo-aulas a aplicações mais complexas, como verificar se tal movimento é humanamente possível e o que deve ser feito para o sucesso de sua execução.

Para que seja possível tal utilização, é necessário o desenvolvimento de uma ferramenta que aproveite as imagens do *Kinect* cujo qual, posteriormente, poderá ser melhorado para aumentar suas funcionalidades e área de atuação. Na *internet* é encontrado, de forma gratuita, um *plug-in* que proporciona a utilização do *Kinect* dentro da plataforma de desenvolvimento *Eclipse*. Esta, que por sua vez, utiliza a linguagem de programação JAVA, usada amplamente em diversos lugares e nos mais variados equipamentos.

Juntamente com o *Kinect*, pode-se encontrar alguns outros equipamentos que também utilizam, ou utilizarão, a captura de movimento, como o *Wavi Xtion* e *PS4 "Orbis"* (IDG, 2013; LIEBL, 2013). Mas, devido à disponibilização do *Framework Microsoft Kinect SDK*, do *plug-in jnect* e um fórum para obtenção de informações, o *Kinect* foi escolhido como a ferramenta a ser explorada por este modelo.

Lembrando que a utilização do *Kinect* em áreas diferentes a de jogos já é realidade. É possível ver alguns exemplos nas notícias de Morales (2013), Souza (2013) e Medicina (2013), onde o *Kinect* é utilizado em diferentes situações dentro da área da medicina. Na área da educação física, tal tecnologia pode ser utilizada para mostrar informações mais minuciosas sobre os exercícios, como por exemplo, postura, quais músculos são trabalhados, posição dos pés e ainda a demonstração do exercício em diferentes ângulos e velocidades sem afetar a qualidade da mesma. Considerando estas praticidades tal modelo poderia ser adequado pra funcionar em diversas modalidades desde atividades mais simples até as mais complexas, como aeróbica e ginástica.

Assim sendo, com a criação de tal ferramenta a criação de vídeo-aulas poderá ficar mais eficiente, devido à possibilidade de alterações durante a criação da animação, tais como qualidade, profundidade das cores, entre outros aspectos. Vale também considerar de que o processo de criação ficará mais veloz uma vez que para a criação básica do vídeo só será que o profissional reproduza o exercício em frente ao *Kinect*. E, futuramente, existe a possibilidade de utilizar a ferramenta para utilizar o *Kinect* também para comparar a animação com a *performance* do usuário, a fim de verificar movimentos excessivos ou desnecessários.

#### 1.4 ESTRUTURA DO TRABALHO

A seguir será apresentada uma breve descrição sobre a organização dos capítulos do trabalho.

O capítulo 2 apresenta os conceitos sobre geometria computacional necessários para identificação das partes do corpo humano, sem a utilização de marcações especiais sobre o corpo do ator. Apresenta também os conceitos de construção e animação de personagens virtuais.

No capítulo 3 são mostradas técnicas para a estimação do movimento humano usando captura óptica de movimento humano, que são utilizadas em trabalhos correlatos.

O capítulo 4 aborda os requisitos identificados, a especificação e a implementação do protótipo. Juntamente com a explicação de algumas partes do seu código.

No capítulo 5 são expostas as conclusões referentes ao trabalho, bem como sugestões para trabalhos futuros.

## **2 DANÇA, CAPTURA DE MOVIMENTO E ANIMAÇÃO COMPUTACIONAL**

Pesquisadores vêm procurando na computação funcionalidades para melhorar ou facilitar a dança. Mas a junção destas áreas aparenta ser algo que possui considerável complexidade, por envolver profissionais de áreas profissionais distintas.

Pesquisadores, no entanto, conseguiram fazer uso da tecnologia para auxiliar nas mais variadas áreas da dança: alguns se destinando à análise de imagens para conversão de uma coreografia em uma notação própria para a dança, de maneira geral na labanotação (PARADISO; HU, 2013, tradução nossa) e, outros, para criar sensores de pressão para avaliar o nível de liberdade durante a dança.

### **2.1 DANÇA**

Presente na sociedade desde a antiguidade, e no reino animal a milhares de anos, a dança é uma forma de exercício que pode possuir os mais variados sentidos. Dentre eles destacam-se diversão, regulagem de peso e melhorarias das habilidades sociais. No passar dos anos já foram desenvolvidas as mais variadas formas para o ensino desta atividade. Começando no ensino de movimentos de dança pelos pais, chegando á simples desenhos das posições dos pés, representações dos movimentos dos corpos, vídeo-aulas ou mesmo com uma escrita própria para a representação de coreografias. No entanto as representações gráficas para o ensino da dança, como animações e imagens, precisam respeitar os limites do corpo humano. Pois todos os movimentos descritos precisam ser possíveis e o mais próximo da realidade possível. Para isso é necessário construir um conhecimento básico referente aos limites do corpo humano e, principalmente, sua representação.

#### **2.1.1 Representação do corpo humano**

Segundo Sales (2013), o corpo humano possui vários tipos de postura, no entanto a sua forma é restrita a uma, tornando mais simples a identificação do corpo humano quando o mesmo estiver em seu alcance. Sales (2013) ainda explica que existem duas formas de modelar um corpo computacionalmente, utilizando a técnica de personagem segmentado ou com a utilização de *metaballs*.

#### 2.1.1.1 Personagem segmentado

Conforme o próprio nome sugere, um personagem segmentado é composto por diversas partes criadas separadamente e depois unidas através de articulações. (SALES, 2013). Devido a isso também se vê necessário o estudo de limites, uma vez que as articulações possuem uma margem para sua movimentação. A tabela 1 mostra os dados referentes às articulações do corpo e seu grau de liberdade.

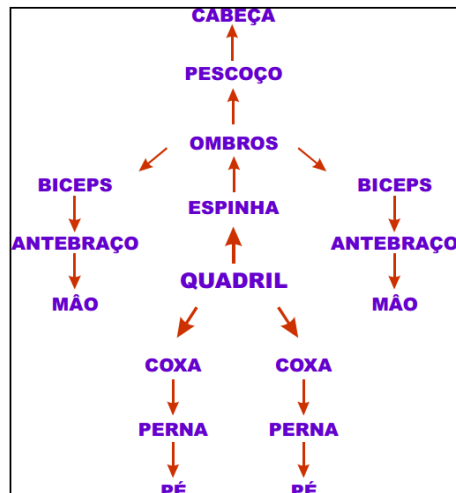
Tabela 1 - Limites rotacionais das articulações

<b>Segmento</b>	<b>Junta</b>	<b>Tipo</b>	<b>X</b>	<b>Y</b>	<b>Z</b>
Pé	Tornozelo	Rotacional	65°	30°	0°
Perna	Joelho	Articulada	135°	0°	0°
Coxa	Quadril	Esfera/Órbita	120°	20°	10
Espinha	Quadril/Espinha	Rotacional	15°	10°	0°
Ombro	Espinha	Rotacional	20°	20°	0°
Bíceps	Ombro	Esfera/Órbita	180°	105°	10°
Antebraço	Cotovelo	Articulada	150°	0°	0°
Mão	Pulso	Esfera/Órbita	180°	30°	120°

Fonte: Sales (2013)

A incorporação de tais limites torna a movimentação das articulações mais realistas, melhorando sua percepção. Mas devido à fisionomia do corpo, as partes não podem se locomover isoladas, tornando necessária a utilização da estrutura de hierarquia do corpo humano, onde é especificada as ligações e ordem de cada membro. (SALES, 2013). A hierarquia pode ser vista detalhadamente na Figura 1.

Figura 1 - Hierarquia do corpo humano

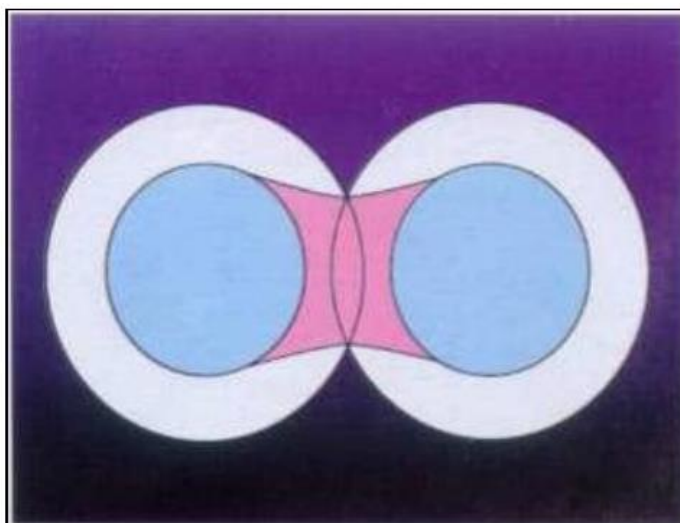


Fonte: Maestri (1996, p. 29) *apud* Sales (2013).

#### 2.1.1.2 Personagem criado com *metaballs*

Utilizadas para a criação de personagens de formato mais arredondado, por se consistir de uma técnica que utiliza esferas com peso e raios de influência individuais. A Figura 2 mostra duas esferas e seus raios de influência.

Figura 2 - Esferas e seus respectivos raios de influência



Fonte: Sales (2013)

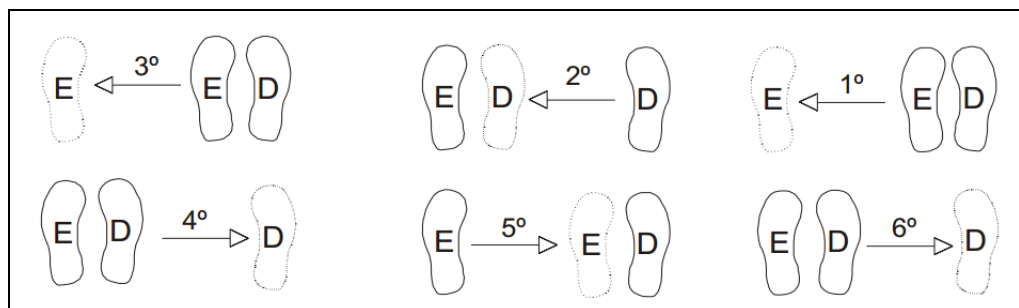
A criação de corpos pode ser feita fundindo diversas esferas, ao mesmo tempo em que seu tamanho, peso e raios de influência são alterados a fim de obterem o formato desejado. (SALES, 2013). No entanto a criação de formas humanas é inviável devido ao elevado número de esferas necessárias, isso ainda combinado com o fato de que a animação destas formas é feita animando cada esfera individualmente. (SALES, 2013).

Com uma compreensão de como o corpo humano se comporta a criação de tutoriais de dança fica extremamente mais fácil. As formas como estes tutoriais são feitos e disponibilizados pode variar, mas de maneira geral eles são feitos com a utilização de imagens ou vídeos.

### 2.1.2 Formas de representação

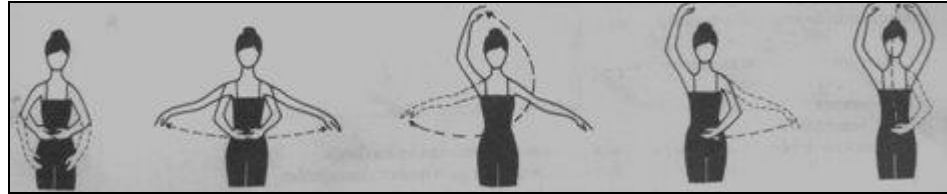
A imagem é uma forma de comunicação muito utilizada no nosso dia-a-dia, como diz o ditado “uma foto vale mais que mil palavras”. Duas formas para o ensino da dança com a utilização de imagens podem ser observadas na Figura 3 e na Figura 4. Sendo que na figura 3 esta sendo representado um simples movimento para se locomover de um lado para o outro, enquanto que na figura 4 é apresentada uma rotina de movimento dos braços no *Ballet*.

Figura 3 - Passos básicos de vaneirão



Fonte: Nunes (2013)

Figura 4 - Posição principais dos braços no Ballet



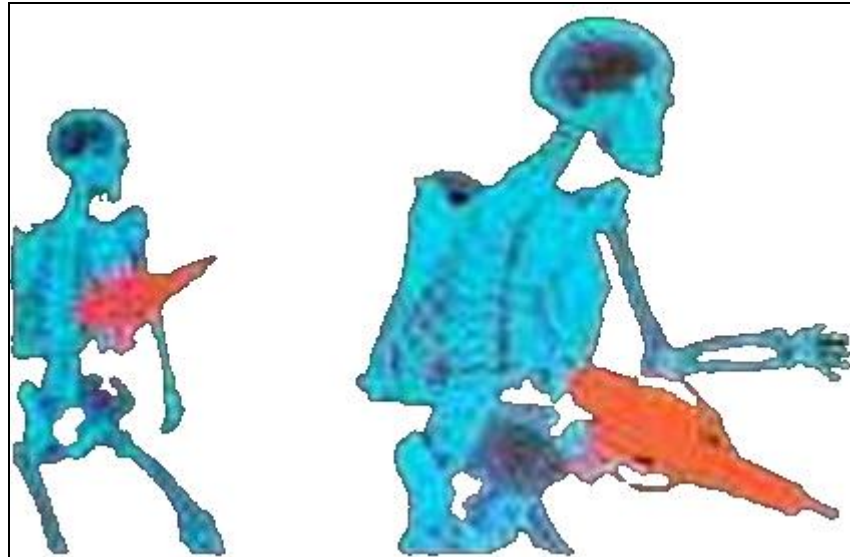
Fonte: Lisboa, et al (2013)

Ainda mais utilizada que a imagem é o recurso de vídeo. Amplamente utilizado nas mais diversas áreas, ele fornece uma maneira extremamente mais visual para o aprendizado da dança, além de oferecer a possibilidade de colocar sons de fundo.

## 2.2 CAPTURA DE MOVIMENTO

Também conhecidos como sistemas de captura e análise de movimentos (SCAM), os sistemas para captura de movimentos possuem como finalidade a identificar e recriar a movimentação de algum objeto no espaço. (SISTO; CABRAL, 2013). Outra definição é dada por Gomide (2013), onde a captura de movimento seria um processo que permite a transcrição de atuação ao vivo em uma atuação digital. Para melhorar o sua compreensão, existem cenas em filmes famosos que são exemplos da aplicação da captura de movimentos. A cena do filme *O Vingador do Futuro*, Figura 5 se aproveitou desta tecnologia para ser elaborada.

Figura 5 – Cena do filme *O Vingador do Futuro*



Fonte: Gomide, et al (2013)

Nesta cena de *O Vingador do Futuro* o personagem principal está sendo perseguido por policiais, e durante a perseguição eles acabam passando por uma máquina de raios-X. Na imagem vemos o que é mostrado pela máquina, a obtenção desta imagem veio graças a um sistema de captura de movimento criada naquela época e ao operador fornecido pelo fabricante (GOMIDE et al, 2013).

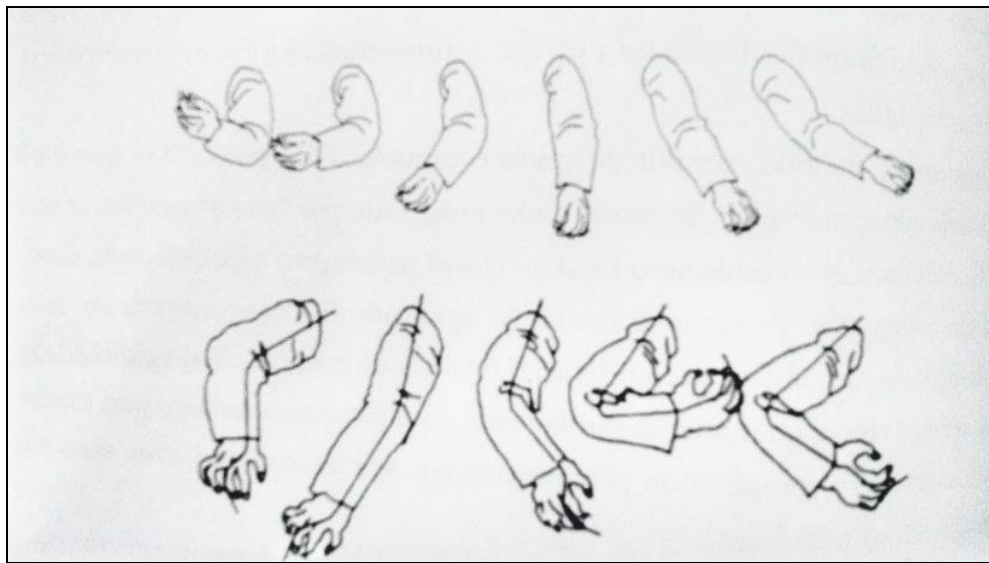
### 2.3 ANIMAÇÃO COMPUTACIONAL

O conceito de animação, segundo Sales (2013), é “a modificação de parâmetros de personagens em um determinado tempo, de tal forma que possamos perceber o movimento de forma natural”. Sendo que a animação computacional nada mais é do que a criação de uma animação utilizando recursos e técnicas computacionais.

As principais técnicas para a criação da animação computacional, ou animação digital, é a utilização de *keyframe* ou a forma direta, frame a frame. (GOMIDE et al, 2013).

A técnica de *keyframe* é similar a forma tradicional para criação de animação, onde o desenhista cria os frames principais (inicial e final) e depois criar os transitivos, que seriam os movimentos feito pelo objeto saindo da posição inicial até a final (LUCENA JR, 2005). A Figura 6 demonstra a aplicação desta técnica.

Figura 6 - Aplicação da técnica de *keyframe*



Fonte: Lucena Jr. (2005)

A animação de direta é feita sequenciando uma cadeia de imagens. Tais imagens podem ser criadas das mais diversas formas e utilizando as mais variadas técnicas de desenho. Sendo apenas necessário que cada quadro da imagem esteja perfeitamente desenhado, para evitar a aparição de erros de sequencia durante a animação. Esta técnica pode ainda ser reproduzida sem a necessidade de um sistema computacional.

#### 2.4 ANIMAÇÃO POR CAPTURA DE MOVIMENTO

Antigamente, desenhistas procuravam estudar detalhadamente os movimentos efetuados por profissionais, para, depois, poder reproduzi-los o mais perfeitamente possível através de desenhos. (PULLEN, 2013, tradução nossa). Tal técnica acabava por ser desgastante, uma vez que o desenhista muitas vezes

precisava ver diversas vezes os movimentos, sem contar o empecilho de que muitos movimentos provavelmente os profissionais não conseguiriam fazer em velocidade reduzida.

Com isso é nítido o benefício da animação com a utilização da captura de movimento. Combinando essas duas tecnologias é possível obter uma animação de forma mais livre e ao mesmo instante em que são feitas pelo profissional. (PULLEN, 2013, tradução nossa).

Diversas são as tecnologias disponíveis para a captura e criação da animação. Sendo que a utilização de um recurso computacional para a obtenção e tratamento dos dados é presente em todas elas. No entanto cada técnica possui pontos fortes e fracos, devendo o pesquisador verificar qual seria a mais adequada para o seu projeto.

### 3 TECNOLOGIA PARA ANIMAÇÃO E CAPTURA DE MOVIMENTO

Diversos são os dispositivos presentes na atualidade capazes de capturar imagens e transformá-las em animação, que nada mais é do que dar vida ao objeto (SALES, 2013). Webcams, filmadoras, celulares, até mesmo simples máquinas fotográficas, com o auxílio de um sequenciador, podem chegar a gerá-las. Diversos são os programas que auxiliam, ou tem como função, na criação de animações ou vídeos. Alguns dos mais referenciados no mundo virtual são o Sony Vegas e Adobe Flash, sendo o primeiro para a criação e edição de vídeos, enquanto que o segundo é destinado a criação de animações em flash.

No entanto para trabalhar com a captura de movimentos, são necessários equipamentos com outra gama de recursos e funcionalidades, capazes de identificar a movimentação de corpos em seu raio de alcance e transmiti-los para um receptor ou equipamento para a exibição e tratamento das imagens . Sistemas acústicos ou de câmeras e sensores de profundidade, inerciais ou eletromagnéticos são exemplos de tais dispositivos. (SISTO; CABRAL, 2013).

#### 3.1 CAPTURA DE MOVIMENTO

Dentre as tecnologias utilizadas para a captura de movimento nos dias atuais, Sisto e Cabral (2013) especificam quais seriam as mais relevantes, podendo estas ser observadas na Tabela 2. E, posteriormente, é apresentado uma descrição mais detalhada sobre cada uma das tecnologias apresentadas na tabela.

Tabela 2 - Sistemas existentes para captura de movimento

SISTEMA	TIPO DE SENSOR	CARACTERÍSTICA	PROBLEMA
Acústico	Sonoro	Utiliza-se da triangulação dos sons para determinar a posição do ator	Sujeito a problemas de reflexão de som e obtenção dos dados em um momento específico

<b>Eletromecânico</b>	Posição angular	Utiliza uma estrutura mecânica com sensores, onde a posição dos sensores é capturada sempre que a mesma se movimenta	A estrutura mecânica dos sensores dificulta a movimentação do ator
<b>Cinéticos</b>	Inerciais	Geralmente utiliza acelerômetros e giroscópios para obter a posição, velocidade linear e angular dos pontos	Atrapalham o movimento do ator
<b>Magnéticos</b>	Magnéticos	Os sensores verificam a posição em 3D do ator, em relação a uma antena transmissora	Interferência decorrente de objetos metálicos próximos
<b>Infravermelho</b>	Infravermelho	Trabalha calculando a distância que os sensores estão da fonte	Possibilidade dos emissores ficarem oclusos no corpo do ator
<b>Óptico-eletrônicos</b>	Câmeras de vídeo e marcadores	As câmeras trabalham capturando o vídeo que é posteriormente analisado por um software para a identificação dos marcadores presente na roupa do ator	Custo muito elevado em decorrência dos materiais necessários

Fonte: Sisto e Cabral (2013)

- a) Sistema acústico: consiste na utilização de emissores e receptores sonoros. Os emissores são ligados ao ator, enquanto que os receptores são colocados na sala. Sequencialmente, os emissores são acionados e os receptores fazem a triangulação dos sons a fim de identificar a posição de cada emissor (SISTO; CABRAL, 2013);
- b) Sistema eletromecânico: utiliza sensores de posição angular que ficam posicionados nas articulações do ator, sendo que quando ele se movimenta os sensores fornecem a posição angular de cada articulação. (SISTO; CABRAL, 2013);
- c) Sistema inercial: geralmente utilizando acelerômetros, este sistema consegue obter a posição e velocidade através de sensores inerciais presentes no ator, mas os dados recebidos dependem dos sensores utilizados. (FLAVEL et al, 2012 *apud* SISTO; CABRAL, 2013);
- d) Sistema magnético: este sistema faz uso de sensores e uma antena que emite sinais de pulso, os sensores recebem os sinais e dão a

posição 3D do ator. Sendo ainda que a eficiência de tal técnica é proporcional à distância dos sensores da antena (RICHARD, 1999 *apud* SISTO; CABRAL, 2013);

- e) Sistema infravermelho: utilizando emissores e receptores, esse sistema faz triangulação da posição dos emissores, acoplados ao ator, sendo sua vantagem a baixa interferência da luz (SISTO; CABRAL, 2013);
- f) Sistema óptico-eletrônico: utilizada amplamente no cinema, este sistema faz uso de marcadores acoplados a roupa do ator. Tais marcadores possuem uma cor de destaque, de forma que um programa de computador possa localiza-los e rastreá-los (SISTO; CABRAL, 2013).

Conforme visto existem diversas maneiras de conseguir capturar o movimento de um corpo. E existem diversas ferramentas que se aproveitaram de tais sistemas, é o caso do *Microsoft Kinect Sensor*, que faz uso de um sistema infravermelho e de câmeras *RGB*.

### 3.1.1 Kinect

Desenvolvido pela *Microsoft*, o *Kinect* é um acessório para o console *Xbox* que possui a finalidade de capturar os movimentos feitos pelo jogador e compará-los com o do jogo ou enviá-lo para que o jogo execute os movimentos. E para tal ele faz uso de uma câmera de profundidade *3D*, uma câmera *RGB*, microfones espalhados pela sua extensão e uma base com inclinação mecanizada (Xbox, 2013; HAN, 2013, tradução nossa), como visto na figura 2.

Figura 7 – Componentes do Microsoft Kinect Sensor



Fonte: Microsoft (2013)

Para facilitar a compreensão, abaixo consta a tradução, ou adaptação, e especificação de cada um dos componentes descritos na imagem:

- a) *3D Depth Sensors* (Sensores de profundidade 3D) - composto por um *laser* e uma câmera de infravermelho, que juntas criam um mapa de profundidade, o que torna possível identificar a distância que o objeto está do aparelho (HAN, 2013, tradução nossa);
- b) *RGB Camera* (Câmera RGB) - responsável pelas três cores básicas da captura de vídeo, com a possibilidade de oferecer imagens com resolução de 640x480 a 30fps ou de 1280x1024 a 10fps. (HAN, 2013, tradução nossa (STRAZDINS; KOMANDUR; STYVE, 2013, tradução nossa);
- c) *Multi-Array Mic* (Microfones alinhados) - o Kinect possui quatro microfones alinhados capazes de capturar a voz do usuário, além de fornecer recursos para redução, ou mesmo anulação, de ruídos e reconhecimento de comandos de voz, em inglês (ROCHA; DEFAVARI; BRANDÃO, 2013);
- d) *Motorized Tilt* (base com inclinação motorizada) - base que pode ter seu ângulo ajustado em 54° diferentes apenas enviando um comando para

o seu motor (STRAZDINS; KOMANDUR; STYVE, 2013, tradução nossa).

Tais componentes e funcionalidades, porém, somente eram aproveitadas pelo console *Xbox*, não sendo possível sua aplicação em outras áreas até o lançamento do *Framework Microsoft Kinect SDK*.

Este *framework* foi lançado pela *Microsoft* para possibilitar, com maior facilidade, a utilização do *Kinect* por desenvolvedores. Ela foi desenvolvida para as linguagens nativas do *Windows*, como C++, C# e *Visual Basic*, no entanto, alguns plug-ins e bibliotecas foram elaborados para a seu aproveitamento em outras linguagens, tal como a linguagem Java. Após a instalação, o usuário ainda tem acesso a exemplos de código e seus executáveis, extremamente úteis para que o desenvolvedor veja as funcionalidades do *Kinect*, fato que pode dar origem a diversas formas de aplicação do *Kinect*, foco deste trabalho é a demonstração de uma delas com a utilização da linguagem Java, plug-ins e bibliotecas.

### 3.2 DESENVOLVIMENTO

Para o desenvolvimento da ferramenta foram utilizados os conceitos estudados referentes à animação, captura de movimento e utilização do *Kinect*, juntamente com a linguagem de programação Java.

Java é uma linguagem muito prática que ganhou muito terreno após seu lançamento, e a cada ano conquista mais afiliados. Além de possuir uma comunidade que auxilia a todos que queira vir a utilizá-la, ela faz uso de uma máquina virtual, ou *virtual machine* (VM), o que a faz funcionar independente de plataforma. Para sua utilização foram estudados dois ambientes, o Eclipse IDE e o Processing.

O Eclipse já é conhecido na área de programação em Java como um dos melhores, e mais recomendados, ambientes de programação. Apesar de inicialmente parecer um pouco complexo sua utilização, com o tempo ele se mostra ser uma ferramenta muito completa, assim como seu “concorrente”, o *Netbeans*.

Infelizmente durante a pesquisa não foi encontrado, inicialmente, um recurso que oferecesse ao Netbeans a possibilidade de comunicação com o Kinect, sendo este substituído por um ambiente mais simples, mas igualmente eficiente.

Desenvolvido inicialmente pelo *Massachusetts Institute of Technology* (Instituto de Tecnologia de Massachussets – MIT), o ambiente *Processing* tinha como finalidade auxiliar no ensinamento de computação gráfica, possuindo uma interface amigável e de fácil interpretação. Hoje ele é utilizado por diversos profissionais ao redor do mundo, devido a seus recursos e praticidade na hora de criar aplicações.

Apesar da qualidade e praticidade dos ambientes estudados, para que o protótipo funcionasse era necessário obter algo com o qual fosse possível a utilização do *Kinect* nestes ambientes. Com isso foram encontrados, inicialmente, um plug-in e duas bibliotecas. Sendo o plug-in destinado ao *Eclipse* e as bibliotecas ao ambiente *Processing*.

O *plug-in* foi encontrado em um site que referenciava o grupo que o desenvolveu. Com um breve descritivo do que foi possível fazer com ele e um vídeo, ele se tornou a primeira ferramenta a ser estudada para que o desenvolvimento do protótipo.

Já as duas bibliotecas são voltadas para o ambiente *Processing*. A *SimpleOpenNI* oferece recursos para utilização do *Kinect* e alguns outros recursos para edição das imagens capturadas. Já o *PKinect* possui mais funções para a captura da imagem do *Kinect* de diferentes maneiras.

A biblioteca *SimpleOpenNI* foi feita com o intuito de tornar mais fácil a utilização dos recursos do *framework OpenNI* pelo ambiente *Processing*. Sendo que este *framework* é utilizado para o desenvolvimento de bibliotecas que auxiliem aplicativos a utilizarem imagens 3D provenientes de dispositivos conectados ao computador (OPENNI, 2013).

Desenvolvida por Bryan Chung, a biblioteca *PKinect* possui recursos práticos para a utilização do *Kinect* no ambiente *Processing*. Sendo notado desde o início de seu desenvolvimento, em Maio de 2012, a preocupação de Chung em mostrar as pessoas que acompanham seu site ([www.magiantdlove.com](http://www.magiantdlove.com)) as novas

funcionalidades da biblioteca, juntamente com um exemplo de código. Código funcional e de fácil compreensão.

O desenvolvimento do protótipo ocorreu na ordem descrita acima, começando pela utilização do *Eclipse* e *Jnect*, passando para o *Processing* e suas bibliotecas logo em seguida.

Em conjunto com todos esses recursos, ainda foi necessária a instalação do Framework Microsoft Kinect *SDK*. Ele é um conjunto de bibliotecas desenvolvido pela *Microsoft* para tornar possível a utilização do *Kinect* por desenvolvedores. Desenvolvida para suportar as linguagens C++, C# e *VisualBasic*, este *framework* vem acompanhado de diversos exemplos de código, juntamente com seus executáveis. Facilitando a execução de testes iniciais e a compreensão, e utilização, de seus recursos.

Já com os recursos acima estudados e analisados, foi possível dar início ao desenvolvimento do protótipo para a geração automática de animação com a utilização do recurso de captura de movimento.

## 4 GERADOR DE VÍDEO A PARTIR DA CAPTURA DE MOVIMENTO

Esta pesquisa consistiu em desenvolver um modelo para geração automática de animação de passos de dança a partir captura de movimentos de dança com a utilização do *Kinect*. Para atingir o objetivo desta pesquisa foram efetuados: uma pesquisa bibliográfica, análise dos componentes indispensáveis para o desenvolvimento do protótipo, localizar e estudar as ferramentas necessárias, elaborar o protótipo, checar os resultados obtidos. Sendo a pesquisa para verificar o que já existe na área referente ao tema, a análise a fim de planejar a obtenção das peças e softwares necessários seguido pelo estudo de seu funcionamento, para somente no final iniciar o desenvolvimento do protótipo e verificar se os resultados foram os esperados.

### 4.1 METODOLOGIA

Dentre as linhas de pesquisa do curso de Ciência da Computação, este projeto se insere no campo de Computação Gráfica. Uma vez que seu objetivo é a criação de um protótipo que irá capturar movimentos reais e transformá-los em uma animação computacional.

As etapas de desenvolvimento da pesquisa foram: levantamento bibliográfico, análise das funcionalidades que o protótipo necessitava ter, criação do modelo de geração automática e testes com as ferramentas de desenvolvimento encontradas.

#### 4.1.1 Levantamento bibliográfico

O desenvolvimento foi baseado nos materiais já produzidos sobre o assunto, como livros, artigos, dissertações, *sites da internet*, projetos e ou produtos comerciais e projetos livres já desenvolvidos. Fontes variadas, como *sites* específicos sobre *Microsoft Kinect*, captura de movimento, modelo esquelético, computação e dança. Mas tanto no que diz respeito de animação, criação de vídeos,

linguagem *Java*, pretendeu-se abordar uma bibliografia que fundamentasse o tema de forma acadêmica.

#### 4.1.2 Análise das funcionalidades

Determinar as funcionalidades que o algoritmo precisa ter foi o primeiro passo a ser dado para definir os recursos que teriam que ser encontrados para o seu desenvolvimento.

Começando pela aparência, que é o que usuário irá ver quando iniciar o protótipo, não se viu necessidade de criar uma tela cheia de funcionalidades, uma vez que o objetivo final é a simples geração de um vídeo. Assim, ficou decidida a criação de uma tela simples onde fosse mostrado, em tempo real, o modelo esquelético do usuário do algoritmo, ou de quem quer que se encontre em frente ao *Kinect*. Lembrando ainda que tal tela deveria ser posteriormente transformada em vídeo através do algoritmo.

A tela necessitava exibir o modelo esquelético do usuário com a utilização do *Kinect*, tornou-se necessário localizar uma maneira de explorar os recursos do *Kinect* através de um aplicativo. Como *Java* é uma das principais linguagens da atualidade, e a que o autor mais domina, a busca se limitou a encontrar maneiras que pudessem ser aproveitadas com a utilização desta linguagem.

Através da busca foi possível encontrar o *plug-in Jnect* e as bibliotecas *SimpleOpenNI* e *PKinect*. Porém, o *plug-in* é destinado para o ambiente *Eclipse*, enquanto que as bibliotecas para o ambiente *Processing*. O ambiente *Eclipse* já possui certa fama no mundo acadêmico, enquanto que o *Processing* é um ambiente não muito difundido, apesar de muito prático e didático.

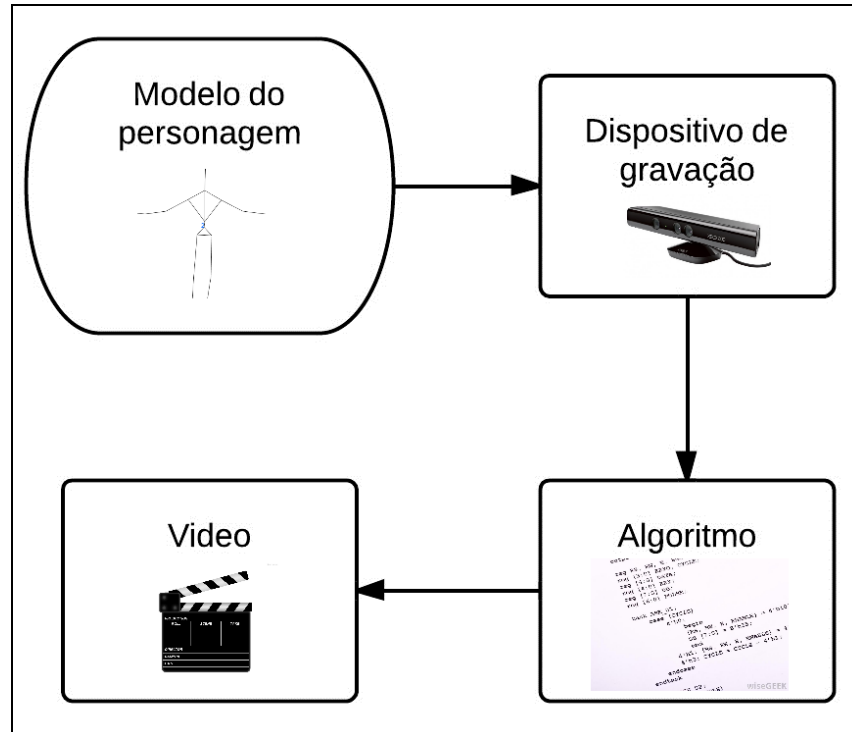
Com os recursos para a obtenção do modelo esquelético já encontrados, é hora de encontrar uma maneira de salvar as imagens mostradas em tela há um arquivo de vídeo. Alguns dos recursos encontrados oferecem tal funcionalidade, sendo possível compreendê-la melhor após o algoritmo estar funcionando apropriadamente, tornando possível o começo do desenvolvimento com a utilização dos recursos encontrados. Mas antes de iniciar qualquer implementação, é

recomendado que o programador faça um esboço do fluxo do programa, o que levou a criação do modelo proposto inicialmente neste trabalho.

#### 4.1.3 Modelo de geração automática

A criação de um modelo é sempre recomendada para auxiliar no entendimento da ferramenta que se pretende desenvolver. Com ele é possível identificar possíveis falhas ou aprimoramentos que necessitam ser feitos antes do seu início, assim como fica mais fácil para quem não está diretamente envolvido com o projeto compreendê-lo. A mostra como ficou a definição do modelo, a explicação de cada quadro pode ser verificada em seguida.

Figura 8 - Modelo para geração automática de animação através da captura de movimento através do *Kinect*



Fonte: do acadêmico, 2013

A primeira etapa consiste na elaboração do modelo do personagem a ser gerado pela captura, sendo decidido escolher uma forma genérica, apenas para fins

de mostrar a movimentação dos membros no protótipo. Assim foi escolhido um modelo que ilustrasse as partes do corpo, interligadas adequadamente para tornar os movimentos o mais real possível.

Porem para a obtenção de tal modelo, é necessário conseguir um dispositivo que ofereça a funcionalidade para captura de movimento. Sendo que existem diversos que oferecem tal recurso, sendo o Kinect o mais popular na atualidade.

Em conjunto com o dispositivo, é necessária a criação de alguma comunicação entre ele e o algoritmo, que torne possível o recebimento dos dados capturados e seu tratamento para a geração do modelo anterior. Sendo que essa geração precisa acontecer em tempo real e salva em algum formato que futuramente pode ser transformado em vídeo.

A transformação em vídeo pode ser feita de diversas maneiras, sendo esta a última etapa da modelagem. A forma como este vídeo será obtido é irrelevante, sendo apenas necessário que seja uma forma automática.

#### **4.1.4 Eclipse e Jnect**

O primeiro *plug-in* estudado foi o *Jnect*, devido a vídeos mostrando suas funções e exemplos de código encontrados, juntamente com o ambiente de desenvolvimento *Eclipse*, uma vez que tal *plug-in* foi desenvolvido especificamente para este ambiente.

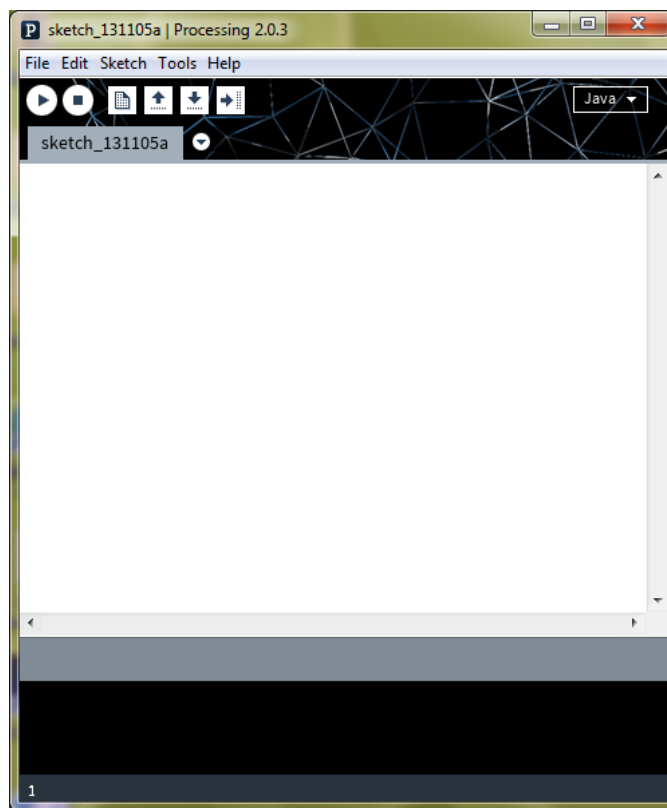
Com a utilização da documentação do *plug-in* e exemplos de código, fazendo uso do mesmo, foi possível obter uma compreensão de como deve ser feita a utilização dele. Seguindo para a compreensão e correta utilização do ambiente *Eclipse*, juntamente com a criação de telas de forma rápida.

Estudando detalhadamente os códigos encontrados fica fácil identificar algumas das funções utilizadas, sem ser necessária a leitura da documentação do *plug-in*. Sendo que para os primeiros testes foram utilizados códigos retirados da própria internet, a fim de verificar se as configurações para que o *plug-in* funcionasse estavam corretas.

#### 4.1.5 Processing e SimpleOpenNI

À primeira vista, o *Processing* se mostrou um ambiente simples para desenvolvimento de ferramentas utilizando a linguagem Java, vide figura 2. Sendo ainda de fácil compreensão e com uma aparência deveras limpa e agradável.

Figura 9 – Tela inicial do ambiente Processing



Fonte: do acadêmico, 2013.

Da mesma forma que o *Jnect*, foram encontrados códigos prontos para utilização do *SimpleOpenNI*, no entanto, esta biblioteca precisava que o *framework OpenNI* estivesse instalado no computador. Após instalação foram utilizados alguns códigos para verificar se a conexão com o *Kinect* iria funcionar. Os testes foram muito positivos, todos os códigos funcionaram perfeitamente.

Foi possível aproveitar partes dos códigos para criar a ferramenta desejada. Após algumas correções e remover algumas linhas que não eram necessárias foi possível ver imagens captadas através do *Kinect*. Após algumas

alterações foi possível utilizar a funcionalidade de captura de movimento para criar o modelo esquelético proposto.

#### 4.1.6 Processing e PKinect

Devido ao pequeno retorno de fontes com a utilização do PKinect, o estudo se limitou a compreensão e análise dos códigos disponibilizados de Bryan Chung ([www.magicandlove.com](http://www.magicandlove.com)). Apesar da pouca quantidade de informações, os códigos encontrados são diretos e de fácil compreensão. Outro fato que contribuiu para a escolha desta biblioteca foi que os códigos encontrados demonstravam a parte de captura de movimento, necessária no algoritmo proposto. A Figura 10 mostra um dos códigos encontrados em execução.

Figura 10 - Exibição 4 formas de captura utilizando *Kinect* e a biblioteca *PKinect*



Fonte: do acadêmico, 2013

Na figura acima podem ser observados quatro tipos de captura diferentes: modelo esquelético, normal, com máscara e profundidade. O modelo esquelético utiliza o recurso de captura de movimento, identificando as partes do corpo e os

transformando em pontos que são interligados posteriormente. A segunda captura é a normal, utilizando a câmera *RGB* do *Kinect*. A captura com máscara funciona capturando o corpo em movimento e sobrepondo-o, caso configurado, a outro fundo, podendo este ser uma imagem, como é o caso do exemplo. A imagem de profundidade é obtida com a utilização da câmera de infravermelho, assim ela apenas captura a distância dos objetos e não sua cor.

Com base no código acima foi possível dar início ao desenvolvimento do algoritmo, necessário para a obtenção e tratamento das imagens capturadas através do *Kinect*. Como já visto o *Processing* faz uso de diversas variáveis globais, como pode ser visto nos códigos e na Figura 11.

Figura 11 - Declaração das variáveis globais da aplicação

```
//declaração das variaveis
PKinect kinect;           //variavel para trabalhar com o o plug-in
ArrayList<SkeletonData> bodies; //variavel para trabalhar com o corpo
boolean captura = false; //variavel para controlar a captura de telas
int i = 1;               //iterador para testes
```

Fonte: do acadêmico, 2013

A variável *bodies* é a responsável por armazenar os dados referentes ao corpo capturado, a variável *captura* é uma *flag* que avisa o programa se é para salvar os frames que estão sendo exibidos ou não, tal variável se mostra necessária devido a configuração da aplicação, onde para iniciar a captura de frames o usuário deve pressionar a tecla “1” e para parar a tecla “2”.

A configuração e criação dos objetos e telas é feita dentro da função *setup()*. Sendo esta acompanhada pela função *draw()*, responsável pelo desenho e atualização da tela. Nela estão localizados a função para desenho do esqueleto, a verificação da *flag captura* e a gravação dos frames apresentados pela aplicação, caso a *flag* esteja com o valor *true*.

A função de desenho do esqueleto funciona criando uma linha entre as partes do esqueleto presentes na biblioteca *PKinect*. Parte do processo de criação dessas linhas pode ser observado na Figura 12.

Figura 12 - Desenho dos ossos do tronco e cabeça

```
//Determinação das partes que serão utilizadas do esqueleto
void drawSkeleton(SkeletonData _s)
{
    // Corpo
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_HEAD,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_CENTER);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_CENTER,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_LEFT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_CENTER,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_RIGHT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_CENTER,
    PKinect.NUI_SKELETON_POSITION_SPINE);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_LEFT,
    PKinect.NUI_SKELETON_POSITION_SPINE);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_RIGHT,
    PKinect.NUI_SKELETON_POSITION_SPINE);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SPINE,
    PKinect.NUI_SKELETON_POSITION_HIP_CENTER);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_HIP_CENTER,
    PKinect.NUI_SKELETON_POSITION_HIP_LEFT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_HIP_CENTER,
    PKinect.NUI_SKELETON_POSITION_HIP_RIGHT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_HIP_LEFT,
    PKinect.NUI_SKELETON_POSITION_HIP_RIGHT);
}
```

Fonte: do acadêmico, 2013

Na figura pode-se perceber a nomenclatura dos pontos a serem interligados através da função *DrawBone*. Começando pela ligação da cabeça ao centro dos ombros, até chegar a conexão da parte esquerda da cintura com a direita. Com o desenho do esqueleto pronto, resta fazer a junção dos frames capturados para a geração do vídeo.

Para esta tarefa foi escolhido o *software ffmpeg*, devido a possibilidade de conversão de imagens em diversos formatos de vídeo de boa qualidade. No entanto ele não é um programa separado e que funciona através de linhas de comando. Portanto para utilizá-lo foi criado um pequeno arquivo “.bat” com o nome

“Conversor”, que precisará ser executada após o final da captura para a criação do vídeo. Completada esta etapa foi dado início aos testes para verificar a qualidade das gravações e se elas alcançaram o objetivo pretendido.

Para os testes foram escolhidos um passo básico de dança e a parte inicial de uma coreografia. O passo escolhido foi o famoso dois-e-dois, representado parcialmente na Figura 3 (pg. 15). Este passo consiste no deslocamento do corpo de um lado para o outro, sendo que ele dará dois passos para um lado e dois para o outro. Já a coreografia foi decidido pegar algo que usasse um pouco os outros membros do corpo, assim foi escolhida o início da coreografia da Zumba® Fitness para a música Marionetta. Tal coreografia pode ser conferida no seguinte *link* (<http://www.youtube.com/watch?v=-3M9DYPMwSI>).

## 4.2 RESULTADOS OBTIDOS

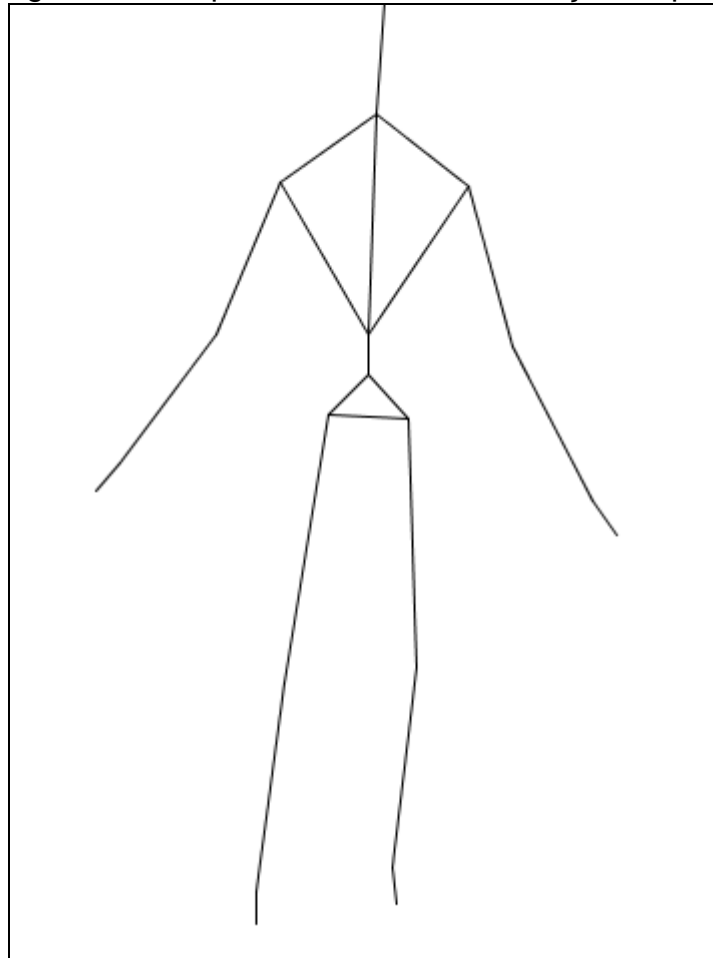
Dentre as três ferramentas analisadas, somente as experiências feitas com a última alcançaram o objetivo deste trabalho. As outras duas apresentaram problemas ou dificuldades que acabaram por atrasar o desenvolvimento do protótipo, motivo que levou a substituição das mesmas.

Com a utilização do *plug-in jnect* infelizmente não foi possível fazer conexão com o *Kinect* durante os testes. Após procurar uma solução para o problema foi verificado que era um erro que realmente estava acontecendo em algumas versões do *plug-in*, principalmente na última, porém o *download* das versões anteriores e da solução para o problema não estavam mais disponíveis. Fato que levou a sua troca pela biblioteca *SimpleOpenNI* e o ambiente *Processing*.

Com eles, a dificuldade foi transformar as imagens capturadas em um arquivo de vídeo. A biblioteca possui uma funcionalidade para gerar um arquivo “vídeo” no formato “oni”, formato lido somente através da utilização do *framework OpenNI*. No entanto não foi possível localizar um meio de transformar esse arquivo em um do tipo vídeo, para ser executado em um player comum no computador. Foram encontradas algumas soluções para trabalhar com os frames, mas não foram encontrados códigos para certificar de que esse método funcionaria.

A biblioteca *PKinect*, no entanto, se mostrou ideal para o cumprimento do objetivo do protótipo. Desde seus primeiros resultados, utilizando os códigos disponibilizados no site de seu criador, ela já mostrou que possuía todos os recursos necessários para, ao menos, conseguir mostrar e capturar os frames recebidos através do *Kinect*. A Figura 13 mostra outro código disponibilizado pelo criador da biblioteca em seu *website*.

Figura 13 – Captura obtida com a utilização do protótipo



Fonte: do acadêmico, 2013

A simplicidade e quantia de material encontrado com a aplicação do *PKinect* reduziram consideravelmente a complexidade no desenvolvimento deste protótipo. Restando somente a conversão das imagens em um arquivo de vídeo, que foi alcançada satisfatoriamente através da aplicação *ffmpeg*.

Os testes feitos com o algoritmo foram muito satisfatórios. Em ambos os passos escolhidos foi possível a identificação dos membros de maneira fácil. No entanto um detalhe chamou a atenção durante os testes. Aparentemente o modelo esquelético pode apresentar alguns problemas caso o usuário faça uso de um cruzamento de membros. Em diversas situações os sensores confundiam os lados quando os braços ou pernas eram cruzados. Fato que acabou por colocar um pequeno limite nos passos que podem ser capturados pelo protótipo.

#### 4.3 APRESENTAÇÃO, DISCUSSÃO E ANÁLISE DOS RESULTADOS OBTIDOS

O motivo principal para desenvolvimento deste protótipo foi fornecer uma base para que futuros pesquisadores pudessem vir a utilizar essa tecnologia com a finalidade de aplica-las a outras áreas que poderiam vir a ser beneficiadas. Sendo o pequeno número de resultados encontrados durante as buscas, para utilização de tal tecnologia, outro fator de extrema relevância que levou a criação desta ferramenta.

Devido ao objetivo deste trabalho ser a criação de um protótipo, há diversas funcionalidades ou modificações que podem ser feitas, como por exemplo, a substituição do modelo esquelético bidimensional por um tridimensional. Substituindo a imagem mostrada de um esqueleto para um personagem mais parecido com um ser humano. Tornando, assim, o algoritmo mais palpável para o ensino de passos ou rotinas de dança. Outra importante funcionalidade considerável seria a de verificar se os movimentos e posturas durante a coreografia estão corretos, ou se não estão forçando demais os músculos. O que pode ser útil se utilizado em um *software* que será repassado ao público, suprimindo parcialmente a necessidade de um profissional capacitado para as atividades físicas oferecidas pelo *software*. Mas este protótipo não está limitado à dança.

Na área da saúde ele pode ser amplamente aproveitado por desportistas, técnicos ou preparadores físicos. Com a obtenção do modelo esquelético poderia surgir a possibilidade de melhorar a performance dos atletas, como postura, inclinação, talvez até a força empregada em tal movimento.

A área de jogos pode ser igualmente favorecida com a utilização e complementação do protótipo. A criação de sequências de movimentos ficaria extremamente mais precisa e muito próxima da realidade. Tornando, talvez, o jogo mais facilmente aceito mesmo para os jogadores mais exigentes. Movimentos limpos, precisos e reais, eis um exemplo do que poderá ser obtido com alguns avanços. Mas não pode ser esquecida a área de animação, que poderá se aproveitar totalmente do algoritmo.

## 5 CONCLUSÃO

Graças às informações organizadas na fundamentação, estudo dos recursos necessários, desenvolvimento do protótipo e análise dos resultados, foi possível identificar e elaborar as seguintes conclusões.

Durante a pesquisa foram percebidas a quantia de possibilidades para a aplicação e aproveitamento da captura de movimento nas mais diversificadas áreas. E sua união com o recurso de animação aumenta consideravelmente sua abrangência, sendo possível verificar sua utilização em estudos da área de educação física, a fim de melhorar o desempenho de desportistas.

Com a utilização das ferramentas *Processing* e da biblioteca *PKinect* foi possível chegar a obtenção e apresentação do modelo esquelético do usuário, sendo que sua animação foi feita com a utilização do *software* *ffmpeg*, resultando em um vídeo de boa qualidade e no formato “.avi”, reconhecido por diversos aparelhos eletrônicos. No vídeo foi possível constatar os movimentos dos membros, que é um fator de extrema importância para o ensino de passos de dança ou mesmo coreografias, onde os movimentos precisam ser precisos.

Juntando e analisando todos os resultados encontrados pode-se confirmar que os objetivos deste trabalho foram alcançados. Podendo através da utilização de um Kinect a obtenção de uma animação do modelo esquelético do usuário em um formato de vídeo. Sendo ainda que tal protótipo pode ser amplamente melhorado.

Melhorias como a substituição dos traços do modelo esquelético por imagens respectivas da parte do corpo ou a inserção da opção de acoplamento de música ao vídeo, já melhorariam consideravelmente a qualidade e eficiência deste protótipo. No entanto a adaptação do protótipo para trabalhar em áreas da saúde poderia resultar em grandes avanços a fim de melhorar a qualidade de vida das pessoas.

## REFERENCIAS

- CHUNG, Bryan. **Magic & Love Interactive**. Disponível em: <<http://www.magicandlove.com/blog/>>. Acessado em: 06 de novembro de 2013.
- CUI, Yan et al. **3D Shape Scanning with a Time-of-Flight Camera**. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.177.9506&rep=rep1&type=pdf>>. Acessado em: 30 ago. 2013.
- EVANS, Don Herbison. **Dance and the Computer: A potential for graphic synergy**. Disponível em: <<http://donhe.topcities.com/pubs/synergy.html>>. Acessado em: 30 ago. 2013.
- GOMIDE, João Victor Boechat, et al. **Captura de movimento e animação de personagens de jogo**. Disponível em: <<http://laplace.dcc.ufmg.br/npdi/uploads/bd254485-3879-77ef.pdf>>. Acessado em: 13 de nov. 2013
- HAN, Jungong; et al. **Enhanced Computer Vision with Microsoft Kinect Sensor: A Review**. Disponível em: <[http://lshao.staff.shef.ac.uk/pub/KinectReview\\_TC2013.pdf](http://lshao.staff.shef.ac.uk/pub/KinectReview_TC2013.pdf)>. Acessado em: 06 nov. 2013.
- IDG NEWS SERVICE. **ASUS demonstra produto semelhante ao Kinect**. Disponível em: <<http://pcworld.uol.com.br/noticias/2011/03/01/asus-demonstra-produto-semelhante-ao-kinect/>>. Acessado em: 30 ago. 2013.
- LIEBL, Matt. **PS4 'Orbis' design sketches suggest Kinect-like device, smartphone Compatibility**. Disponível em: <<http://www.gamezone.com/news/2012/03/29/ps4-orbis-design-sketches-suggest-kinect-like-device-smartphone-compatibility>>. Acessado em: 30 ago. 2013.
- LISBOA, et al. **Metodologia do ensino da dança na educação física "A origem do Ballet"**. Disponível em: <<http://educfisicaufal.files.wordpress.com/2010/12/a-origem-do-ballet.pdf>>. Acessado em: 13 de nov. de 2013
- LUCENA JR, Alberto. **Arte da animação: técnica e estética através da história**. 2ª ed. São Paulo: Senac, 2009. 456p.
- MAESTRI, George. **Animação [digital] de personagens**. São Paulo: Quark do Brasil Ltda., 1996.
- MEDICINA EM GOIÁS. **Lesões ortopédicas são tratadas com recursos virtuais**. Disponível em: <<http://www.medicinago.com.br/materia/9-lesoes-ortopedicas-podem-ser-tratadas-com-recursos-virtuais>>. Acessado em: 30 ago. 2013.

MICROSOFT Research. **Programming with Kinect to Windows SDK**. Disponível em: <[http://research.microsoft.com/en-us/events/fs2011/jancke\\_kinect\\_programming.pdf](http://research.microsoft.com/en-us/events/fs2011/jancke_kinect_programming.pdf)>. Acessado em: 06 nov. 2013.

MORALES, Fernanda. **Hospital brasileiro usa o Kinect durante cirurgias para visualização de exames**. Disponível em: <<http://corporate.canaltech.com.br/noticia/saude/Hospital-brasileiro-usa-Kinect-da-Microsoft-durante-cirurgias/>>. Acessado em: 30 ago. 2013.

MOREIRA, Marilda Maria da Silva. **Qualidade de Vida: Expressões Subjetivas e Histórico-Sociais**. Disponível em: <[http://www.uel.br/revistas/ssrevista/cv9n1\\_marilda.htm](http://www.uel.br/revistas/ssrevista/cv9n1_marilda.htm)>. Acessado em: 30 ago. 2013.

NUNES, Vini. **Manual de danças gaúchas**. Disponível em: <[http://www.aprendadancar.xpg.com.br/Manual\\_Dancas\\_de\\_Salao.pdf](http://www.aprendadancar.xpg.com.br/Manual_Dancas_de_Salao.pdf)>. Acessado em: 13 de nov. de 2013

OPENNI. **Sobre o OpenNI**. Disponível em: <<http://www.openni.org/>>. Acessado em: 09 de set. de 2013

PARADISO, Joseph A.; HU, Eric. **Expressive Footwear for Computer-Augmented Dance Performance**. Disponível em: <[http://reserv.media.mit.edu/pubs/papers/97\\_10\\_Wearcon\\_Shoe.pdf](http://reserv.media.mit.edu/pubs/papers/97_10_Wearcon_Shoe.pdf) >. Acessado em: 06 nov. 2013.

PREVIDÊNCIA USIMINAS. **Educação Financeira e Previdenciária: Artigos relacionados**. Disponível em: <[http://www.previdenciausiminas.com/educacao\\_financeira\\_previdenciaria/artigos\\_relacionados/lazer\\_diversao\\_terceira\\_idade.php](http://www.previdenciausiminas.com/educacao_financeira_previdenciaria/artigos_relacionados/lazer_diversao_terceira_idade.php)>. Acessado em: 30 ago. 2013.

PULLEN, Katherine Ann. **Motion capture assisted animation: texturing and synthesis**. Disponível em: <<http://nccastaff.bournemouth.ac.uk/hncharif/MathsCGs/Desktop/Research/General%20Papers/thesisD11.pdf>>. Acessado em: 13 de nov. de 2013.

QUEIROZ, Murilo. **Um cientista explica o Microsoft Kinect**. Disponível em: <<http://blog.vettalabs.com/2010/10/29/um-cientista-explica-o-microsoft-kinetic/>>. Acessado em: 30 ago. 2013.

ROCHA, Pollyeverlin R.; DEFAVARI, Alex H.; BRANDÃO, Pierre S.. **Estudo da viabilidade da utilização do Kinect como ferramenta no atendimento fisioterapêutico de pacientes neurológicos**. Disponível em: <<http://sbgames.org/sbgames2012/proceedings/papers/gamesforchange/g4c-04.pdf>>. Acessado em: 06 nov. 2013.

SALES, Silvia Hedla Correia de. **Animação de uma Marionete Virtual, a partir de um Sistema Óptico de Captura de Movimento Humano, sem Utilização de**

**Marcações Especiais.** Disponível em: <<http://campeche.inf.furb.br/tccs/2004-II/2004-2silviahcsalesvf.pdf>>. Acessado em: 12 de nov. de 2013

SALVE, Mariângela Gagliardi Caro. **A prática da atividade física:** Estudo comparativo entre os alunos de graduação da UNICAMP. Disponível em: <<http://www.scielo.gpeari.mctes.pt/pdf/mot/v4n3/v4n3a06.pdf>>. Acessado em: 30 ago. 2013.

SCHMIDT, Mirko; JÄHNE, Bernd. **A Physical Model of Time-of-Flight 3D Imaging Systems, Including Suppression of Ambient Light.** Disponível em: <[http://hci.iwr.uni-heidelberg.de/Staff/mischmid/pdf/schmidt2009\\_ToFModel\\_Dyn3D.pdf](http://hci.iwr.uni-heidelberg.de/Staff/mischmid/pdf/schmidt2009_ToFModel_Dyn3D.pdf)>. Acessado em: 30 ago. 2013.

SISTO, Tiago Miziara; CABRAL, Eduardo Lobo. **Sistema de captura de movimentos com sensores infravermelho.** Disponível em: <<http://www.maua.br/pesquisas/sistema-de-captura-de-movimento-com-sensores-infravermelho>>. Acessado em 12 de nov. de 2013

SOUZA, Leandro. **Software transforma Kinect em terapeuta virtual.** Disponível em: <<http://www.baguete.com.br/noticias/05/04/2013/software-transforma-kinect-em-terapeuta-virtual>>. Acessado em: 06 nov. 2013.

STRAZDINS, Girts; KOMANDUR, Sashidharan; STYVE, Arne. **Kinect-based systems for maritime operation simulators?** Disponível em: <[http://www.scs-europe.net/dlib/2013/ecms13papers/ese\\_ECMS2013\\_0041.pdf](http://www.scs-europe.net/dlib/2013/ecms13papers/ese_ECMS2013_0041.pdf)>. Acessado em: 06 nov. 2013.

XBOX. **Componentes do Sensor Kinect.** Disponível em: <<http://support.xbox.com/pt-BR/xbox-360/kinect/kinect-sensor-components>>. Acessado em: 06 nov. 2013.

## REFERENCIAS COMPLEMENTARES

AKSOY, Gokhan. **Effect of Computer Animation Technique on Students' Comprehension of the "Solar System and Beyond" Unit in the Science and Technology Course.** Disponível em: <<http://files.eric.ed.gov/fulltext/ED543588.pdf> >. Acessado em: 06 nov. 2013.

BERG, Tamara; et al. **Interactive Music: Human Motion Initiated Music Generation Using Skeletal Tracking By Kinect.** Disponível em: <[http://www.tamaraberg.com/papers/kinect\\_music.pdf](http://www.tamaraberg.com/papers/kinect_music.pdf) >. Acessado em: 06 nov. 2013.

DUNN, Meghan. **Technology in the courtroom: An examination of the effects of computer animation.** Disponível em: <[http://www.fjc.gov/public/pdf.nsf/lookup/CTtech04.pdf/\\$file/CTtech04.pdf](http://www.fjc.gov/public/pdf.nsf/lookup/CTtech04.pdf/$file/CTtech04.pdf) >. Acessado em: 06 nov. 2013.

**Film and video: history and technical aspects.** Disponível em: <[http://www.mhprofessional.com/downloads/products/0071791361/0071791361\\_chap01.pdf](http://www.mhprofessional.com/downloads/products/0071791361/0071791361_chap01.pdf)>. Acessado em: 06 nov. 2013.

FORTIUM GRUPO EDUCACIONAL. **História do Java.** Disponível em: <[http://fortium.edu.br/blog/vanderlei\\_allles/files/2011/02/Apostila\\_Java\\_atual1.pdf](http://fortium.edu.br/blog/vanderlei_allles/files/2011/02/Apostila_Java_atual1.pdf)>. Acessado em: 06 nov. 2013.

GALL, et al. **Motion capture using joint skeleton tracking and surface estimation.** Disponível em: <<http://www.mpi-inf.mpg.de/~stoll/paper/stse.pdf>>. Acessado em: 06 nov. 2013.

GONTIJO, Silvana. **O livro de ouro da comunicação.** Rio de Janeiro: Ediouro, 2004. 463p.

JANA, Abhijit. **Kinect for Windows SDK Programming Guide.** Birmingham: Packt Publishing, 2012. 392p.

JNECT. Disponível em: <<https://code.google.com/a/eclipselabs.org/p/jnect/>>. Acessado em: 06 de novembro de 2013

MACHADO, Daniel Ira; SANTOS, Plácida L. V. Amorim da Costa. **Avaliação da hipermídia no processo de ensino e aprendizagem da física: o caso da gravitação.** Disponível em: <<http://www.scielo.br/pdf/ciedu/v10n1/06.pdf> >. Acessado em: 06 nov. 2013.

MORAES, Marcelo. **A história do surgimento da linguagem JAVA.** Disponível em: <[http://www.marcelomoraes.com.br/conteudo/marcelo/java/historia\\_java.pdf](http://www.marcelomoraes.com.br/conteudo/marcelo/java/historia_java.pdf)>. Acessado em: 06 nov. 2013.

MOURA, João Martinho; et al. **You Move You Interact: a full-body dance in-between reality and virtuality**. Disponível em: <[https://repositorioaberto.uab.pt/bitstream/10400.2/2297/1/YMYI\\_Artech.pdf](https://repositorioaberto.uab.pt/bitstream/10400.2/2297/1/YMYI_Artech.pdf)>. Acessado em: 06 nov. 2013.

PAGAN, Manuela. **Leve em consideração sete critérios antes de escolher uma atividade física**. Disponível em: <<http://www.minhavidacom.br/fitness/galerias/16372-leve-em-consideracao-sete-criterios-antes-de-escolher-uma-atividade-fisica>>. Acessado em: 30 ago. 2013.

PARADISO, J. A.; SPARACINO, F.. **Optical Tracking for Music and Dance Performance**. Disponível em: <[http://www.cba.mit.edu/docs/papers/97.09.Zurich\\_3D4.pdf](http://www.cba.mit.edu/docs/papers/97.09.Zurich_3D4.pdf) >. Acessado em: 06 nov. 2013.

PINTO, Aline da Silva. **Dança e educação: a escola como espaço para a construção de relações de gênero mais igualitárias e de respeito às diferenças**. Disponível em: <<http://www.unilasalle.edu.br/canoas/assets/upload/ALINE%20V%20FINAL.pdf>>. Acessado em: 12 de nov. de 2013.

PROCESSING. Disponível em: <<http://processing.org/>>. Acessado em: 06 de novembro de 2013

PULLEN, Katherine; BREGLER, Christoph. **Motion capture assisted animation: texturing and synthesis**. Disponível em: <<http://cims.nyu.edu/~bregler/papers375.pdf>>. Acessado em: 13 de nov. de 2013

SENTÜRK, Sertan; et al. **Crossole: A Gestural Interface for Composition, Improvisation and Performance using Kinect**. Disponível em: <[http://www.eecs.umich.edu/nime2012/Proceedings/papers/185\\_Final\\_Manuscript.pdf](http://www.eecs.umich.edu/nime2012/Proceedings/papers/185_Final_Manuscript.pdf)>. Acessado em: 06 nov. 2013.

SUKEL, Katherine E.; et al. **Presenting Movement in a Computer-Based Dance Tutor**. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=C71EC26C29675F9BCA7B4305D7CE3F59?doi=10.1.1.86.6239&rep=rep1&type=pdf> >. Acessado em: 06 nov. 2013.

**The History and Evolution of Java**. Disponível em: <[http://books.mcgraw-hill.com/downloads/products/0072230738/0072230738\\_c01.pdf](http://books.mcgraw-hill.com/downloads/products/0072230738/0072230738_c01.pdf)>. Acessado em: 06 nov. 2013.

WANG, Xiaoting; WANG, Lu; WU, Guosheng. **Body and face animation based on motion capture**. Disponível em: <<http://www.mecspress.org/ijieeb/ijieeb-v3-n2/IJIEEB-V3-N2-4.pdf>>. Acessado em: 09 set. 2013.

WATERS, Keith; TERZOPOULOS, Demetri. **The Computer Synthesis of Expressive Faces**. Disponível em: <<http://69.89.31.178/~pilgrimw/ifacelab/docs/rs92.pdf> >. Acessado em: 06 nov. 2013.

ZORDAN, Victor B.; CELLY ,Bhrigu; CHIU, Bill; DILORENZOY , Paul C.. **Breathe Easy: Model and control of simulated respiration for animation**. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.10.5214&rep=rep1&type=pdf>>. Acessado em: 06 nov. 2013.

**ANEXOS**

## ANEXO A – Código fonte do protótipo

```

import pKinect.PKinect;

//declaração das variáveis
PKinect kinect;           //variavel para trabalhar com o plug-in
ArrayList<SkeletonData> bodies; //variavel para trabalhar com o corpo
boolean captura = false; //variavel para controlar a captura de telas
int i = 1;                //iterador para testes

//configuração
void setup()
{
    size(640, 480);           //resolução
    kinect = new PKinect(this); //criação do objeto kinect
    bodies = new ArrayList<SkeletonData>(); // criação do objeto bodies
    smooth();
}

//função de desenho
void draw()
{
    background(255);           //cor do plano de fundo
    for (int i=0; i<bodies.size(); i++)
    {

        //desenhar esqueleto
        drawSkeleton(bodies.get(i));
    }

    //Teste para verificar se a captura de tela esta ligada
    if (captura) {
        if (i<10) {
            saveFrame("Telas/000"+i+".png"); //função para salvar o frame atual em disco
        }

        //Tests para nomear as telas capturadas
        else {
            if (i<100) {
                saveFrame("Telas/00"+i+".png");
            }
            else {
                if (i<1000) {
                    saveFrame("Telas/0"+i+".png");
                }
                else {
                    if (i<10000) {
                        saveFrame("Telas/"+i+".png");
                    }
                }
            }
        }
    }
    i++;
}

//evento para inicio e pausa da captura de tela
void keyPressed()
{
    if (key == '1') {
        captura = true;
    }
}

```

```

}
else {
    if (key == '2') {
        captura = false;
    }
}
}
}

//Determinação das partes que serão utilizadas do esqueleto
void drawSkeleton(SkeletonData _s)
{
    // Corpo
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_HEAD,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_CENTER);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_CENTER,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_LEFT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_CENTER,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_RIGHT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_CENTER,
    PKinect.NUI_SKELETON_POSITION_SPINE);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_LEFT,
    PKinect.NUI_SKELETON_POSITION_SPINE);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_RIGHT,
    PKinect.NUI_SKELETON_POSITION_SPINE);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SPINE,
    PKinect.NUI_SKELETON_POSITION_HIP_CENTER);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_HIP_CENTER,
    PKinect.NUI_SKELETON_POSITION_HIP_LEFT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_HIP_CENTER,
    PKinect.NUI_SKELETON_POSITION_HIP_RIGHT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_HIP_LEFT,
    PKinect.NUI_SKELETON_POSITION_HIP_RIGHT);

    // Braço esquerdo
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_LEFT,
    PKinect.NUI_SKELETON_POSITION_ELBOW_LEFT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_ELBOW_LEFT,
    PKinect.NUI_SKELETON_POSITION_WRIST_LEFT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_WRIST_LEFT,
    PKinect.NUI_SKELETON_POSITION_HAND_LEFT);

    // Braço direito
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_SHOULDER_RIGHT,
    PKinect.NUI_SKELETON_POSITION_ELBOW_RIGHT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_ELBOW_RIGHT,
    PKinect.NUI_SKELETON_POSITION_WRIST_RIGHT);
    DrawBone(_s,
    PKinect.NUI_SKELETON_POSITION_WRIST_RIGHT,

```

```

PKinect.NUI_SKELETON_POSITION_HAND_RIGHT);

// Perna Esquerda
DrawBone(_s,
PKinect.NUI_SKELETON_POSITION_HIP_LEFT,
PKinect.NUI_SKELETON_POSITION_KNEE_LEFT);
DrawBone(_s,
PKinect.NUI_SKELETON_POSITION_KNEE_LEFT,
PKinect.NUI_SKELETON_POSITION_ANKLE_LEFT);
DrawBone(_s,
PKinect.NUI_SKELETON_POSITION_ANKLE_LEFT,
PKinect.NUI_SKELETON_POSITION_FOOT_LEFT);

// Perna Direita
DrawBone(_s,
PKinect.NUI_SKELETON_POSITION_HIP_RIGHT,
PKinect.NUI_SKELETON_POSITION_KNEE_RIGHT);
DrawBone(_s,
PKinect.NUI_SKELETON_POSITION_KNEE_RIGHT,
PKinect.NUI_SKELETON_POSITION_ANKLE_RIGHT);
DrawBone(_s,
PKinect.NUI_SKELETON_POSITION_ANKLE_RIGHT,
PKinect.NUI_SKELETON_POSITION_FOOT_RIGHT);
}

//Desenho dos ossos
void DrawBone(SkeletonData _s, int _j1, int _j2)
{
    noFill();
    stroke(0, 0, 0);
    if (_s.skeletonPositionTrackingState[_j1] != PKinect.NUI_SKELETON_POSITION_NOT_TRACKED &&
_s.skeletonPositionTrackingState[_j2] != PKinect.NUI_SKELETON_POSITION_NOT_TRACKED) {
        line(_s.skeletonPositions[_j1].x*width,
_s.skeletonPositions[_j1].y*height,
_s.skeletonPositions[_j2].x*width,
_s.skeletonPositions[_j2].y*height);
    }
}

//Evento
void appearEvent(SkeletonData _s)
{
    if (_s.trackingState == PKinect.NUI_SKELETON_NOT_TRACKED)
    {
        return;
    }
    synchronized(bodies) {
        bodies.add(_s);
    }
}

//evento para o esqueleto desaparecer
void disappearEvent(SkeletonData _s)
{
    synchronized(bodies) {
        for (int i=bodies.size()-1; i>=0; i--)
        {
            if (_s.dwTrackingID == bodies.get(i).dwTrackingID)
            {
                bodies.remove(i);
            }
        }
    }
}

```

```
}  
  
//evento para movimento do esqueleto  
void moveEvent(SkeletonData _b, SkeletonData _a)  
{  
    if (_a.trackingState == PKinect.NUI_SKELETON_NOT_TRACKED)  
    {  
        return;  
    }  
    synchronized(bodies) {  
        for (int i=bodies.size()-1; i>=0; i--)  
        {  
            if (_b.dwTrackingID == bodies.get(i).dwTrackingID)  
            {  
                bodies.get(i).copy(_a);  
                break;  
            }  
        }  
    }  
}
```

## APÊNDICE

## APENDICE A - Artigo

### **Animação por captura de movimento com o uso do Kinect: uma aplicação para instrução de dança**

**Bruno Nunes Bittencourt<sup>1</sup>, Leila Laís Gonçalves<sup>2</sup>**

<sup>1</sup>Acadêmico no curso de Ciência da Computação - Universidade do Extremo Sul Catarinense (UNESC) – Criciúma – SC – Brazil

<sup>2</sup>Professora no curso de Ciência da Computação – Universidade do Extremo Sul Catarinense (UNESC) – Criciúma – SC – Brazil

bruginho@hotmail.com, llg@unesc.net

***Abstract.** The applicability of the motion capture has been explored for several areas. Being possible, its use to assist in the magnification of accessibility and quality of routines and exercises. For this to happen, a study is necessary about the modeling of the human body, the creation of a model for understanding the flow of information and the execution of tests in order to determine its efficiency. In possession of this information become possible the creation of a prototype, which uses the Kinect tool to motions capture and its conversion in a computer animation.*

***Resumo.** As aplicabilidades da captura de movimento vêm sendo exploradas pelas mais diversas áreas. Sendo possível a sua utilização para auxiliar na ampliação de acessibilidade e qualidade de rotinas e exercícios físicos. Para que isso ocorra é necessário um estudo sobre a modelagem do corpo humano, a criação de um modelo para compreensão do fluxo de informações e a execução de testes para determinar sua eficiência. Em posse destas informações foi possível a criação de um protótipo para a utilização da ferramenta Kinect para a captura de movimentos e sua conversão em uma animação digital.*

#### **Introdução**

A melhoria na qualidade de vida tem sido, nos últimos tempos, uma busca contínua levando as pessoas a mudanças na alimentação, rotina, lugares frequentados e adesão a algum tipo de atividade física (MOREIRA, 2013). As atividades físicas disponíveis são diversas e dentre as mais procuradas destacam-se a musculação, caminhadas, andar de bicicleta, natação, ginástica e dança (PREVIDÊNCIA, 2013; SALVE, 2013).

Na busca por atividade orientada por profissionais, as pessoas se deparam com diferentes problemas na adesão. Um desses problemas é o encaixe da atividade em sua rotina referente aos horários de oferta. Apesar da flexibilidade dos horários, já possibilitada por academias ou outros estabelecimentos, nem todas as modalidades de atividades estão disponíveis.

Assim este artigo apresenta um protótipo que procura minimizar o problema apresentado. Para isso ele faz a utilização do *Kinect* para a captura dos movimentos, de bibliotecas de um framework de desenvolvimento (por exemplo, *OpenKinect*, *OpenNI* ou *Microsoft Kinect SDK*) e de *plug-ins* para uso dos movimentos capturados (por exemplo, o *plug-in jnect*). Estes testados nos ambientes de desenvolvimento Eclipse e Processing.

## Representação do corpo humano

Antes de dar início a pesquisa para a criação do modelo computacional, foi decidido estudar a estrutura do corpo humano, uma vez que será utilizado um personagem computacional para a demonstração dos movimentos. Foi possível encontrar documentos com informações referentes a criação gráfica de corpos humanos com a utilização, técnica referenciada como “representação do corpo humano”.

Segundo Sales (2013), o corpo humano possui vários tipos de postura, no entanto a sua forma é restrita a uma, tornando mais simples a identificação do corpo humano quando o mesmo estiver em seu alcance. Sales (2013) ainda explica que existem duas formas de modelar um corpo computacionalmente, utilizando a técnica de personagem segmentado ou com a utilização de *metaballs*.

## Personagem segmentado

Conforme o próprio nome sugere, um personagem segmentado é composto por diversas partes criadas separadamente e depois unidas através de articulações. (SALES, 2013). Devido a isso também se vê necessário o estudo de limites, uma vez que as articulações possuem uma margem para sua movimentação. A tabela 1 mostra os dados referentes às articulações do corpo e seu grau de liberdade.

Segmento	Junta	Tipo	X	Y	Z
Pé	Tornozelo	Rotacional	65°	30°	0°
Perna	Joelho	Articulada	135°	0°	0°
Coxa	Quadril	Esfera/Órbita	120°	20°	10
Espinha	Quadril/Espinha	Rotacional	15°	10°	0°
Ombro	Espinha	Rotacional	20°	20°	0°
Bíceps	Ombro	Esfera/Órbita	180°	105°	10°
Antebraço	Cotovelo	Articulada	150°	0°	0°
Mão	Pulso	Esfera/Órbita	180°	30°	120°

**Tabela 3 - Limites rotacionais das articulações**

A incorporação de tais limites torna a movimentação das articulações mais realistas, melhorando sua percepção. Mas devido à fisionomia do corpo, as partes não podem se locomover isoladas, tornando necessária a utilização da estrutura de hierarquia do corpo humano, onde é especificada as ligações e ordem de cada membro. (SALES, 2013). A hierarquia pode ser vista detalhadamente na Figura 1.

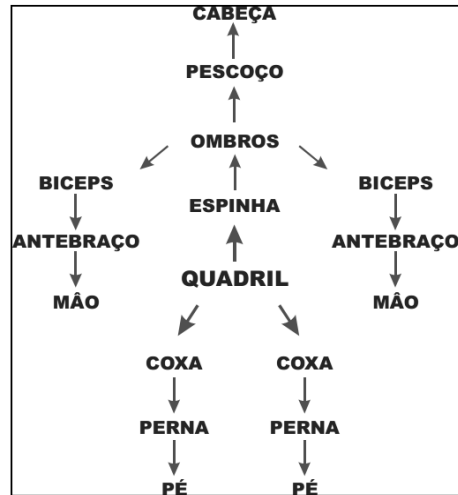


Figura 14 - Hierarquia do corpo humano

### Personagem criado com *metaballs*

Utilizadas para a criação de personagens de formato mais arredondado, por se consistir de uma técnica que utiliza esferas com peso e raios de influência individuais. A Figura 2 mostra duas esferas e seus raios de influência.

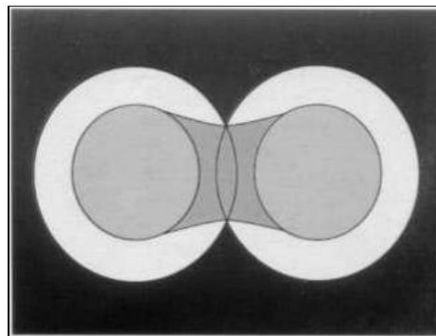


Figura 15 - Esferas e seus respectivos raios de influência

A criação de corpos pode ser feita fundindo diversas esferas, ao mesmo tempo em que seu tamanho, peso e raios de influência são alterados a fim de obterem o formato desejado. (SALES, 2013). No entanto a criação de formas humanas é inviável devido ao elevado número de esferas necessárias, isso ainda combinado com o fato de que a animação destas formas é feita animando cada esfera individualmente. (SALES, 2013).

Independente da técnica escolhida, ainda existirá certa complexidade em animar estas estruturas. Assim a escolha da técnica de “captura de movimento” se mostrou ideal para uma maior agilidade na obtenção e animação de gestos feitos por um ator.

## Captura de movimento

Também conhecidos como sistemas de captura e análise de movimentos (SCAM), os sistemas para captura de movimentos possuem como finalidade a identificar e recriar a movimentação de algum objeto no espaço. (SISTO; CABRAL, 2013). Outra definição é dada por Gomide (2013), onde a captura de movimento seria um processo que permite a transcrição de atuação ao vivo em uma atuação digital. Para melhorar o sua compreensão, existem cenas em filmes famosos que são exemplos da aplicação da captura de movimentos. Uma das cenas do filme *O Vingador do Futuro*, Figura 53 mostra como a captura de movimento já foi, e continua sendo, utilizada na área cinematográfica.

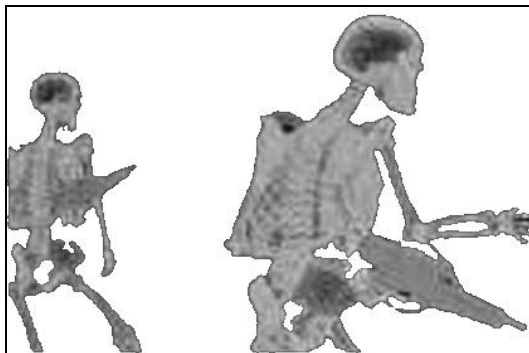


Figura 16 – Cena do filme *O Vingador do Futuro*

Nesta cena de *O Vingador do Futuro* o personagem principal está sendo perseguido por policiais, e durante a perseguição eles acabam passando por uma máquina de raios-X. Na imagem vemos o que é mostrado pela máquina, sendo que a obtenção da animação veio graças a um sistema de captura de movimento criada naquela época e ao operador fornecido pelo fabricante (GOMIDE et al, 2013).

## Animação por captura de movimento

Antigamente, desenhistas procuravam estudar detalhadamente os movimentos efetuados por profissionais, para, depois, poder reproduzi-los o mais perfeitamente possível através de desenhos. (PULLEN, 2013, tradução nossa). Tal técnica acabava por ser desgastante, uma vez que o desenhista muitas vezes precisava ver diversas vezes os movimentos, sem contar o empecilho de que muitos movimentos provavelmente os profissionais não conseguiriam fazer em velocidade reduzida.

Com isso é nítido o benefício da animação com a utilização da captura de movimento. Combinando essas duas tecnologias é possível obter uma animação de forma mais livre e ao mesmo instante em que são feitas pelo profissional. (PULLEN, 2013, tradução nossa).

Diversas são as tecnologias disponíveis para a captura e criação da animação. Sendo que a utilização de um recurso computacional para a obtenção e tratamento dos dados é presente em todas elas. No entanto cada técnica possui pontos fortes e fracos, devendo o pesquisador verificar qual seria a mais adequada para o seu projeto.

## Tecnologias para a captura de movimento

Dentre as tecnologias utilizadas para a captura de movimento nos dias atuais, Sisto e Cabral (2013) especificam quais seriam as mais relevantes, podendo estas ser observadas na Tabela 2.

<b>SISTEMA</b>	<b>TIPO DE SENSOR</b>	<b>CARACTERÍSTICA</b>	<b>PROBLEMA</b>
<b>Acústico</b>	Sonoro	Utiliza-se da triangulação dos sons para determinar a posição do ator	Sujeito a problemas de reflexão de som e obtenção dos dados em um momento específico
<b>Eletromecânico</b>	Posição angular	Utiliza uma estrutura mecânica com sensores, onde a posição dos sensores é capturada sempre que a mesma se movimenta	A estrutura mecânica dos sensores dificulta a movimentação do ator
<b>Cinéticos</b>	Inerciais	Geralmente utiliza acelerômetros e giroscópios para obter a posição, velocidade linear e angular dos pontos	Atrapalham o movimento do ator
<b>Magnéticos</b>	Magnéticos	Os sensores verificam a posição em 3D do ator, em relação a uma antena transmissora	Interferência decorrente de objetos metálicos próximos
<b>Infravermelho</b>	Infravermelho	Trabalha calculando a distância que os sensores estão da fonte	Possibilidade dos emissores ficarem oclusos no corpo do ator
<b>Óptico-eletrônicos</b>	Câmeras de vídeo e marcadores	As câmeras trabalham capturando o vídeo que é posteriormente analisado por um software para a identificação dos marcadores presente na roupa do ator	Custo muito elevado em decorrência dos materiais necessários

**Tabela 4 - Sistemas existentes para captura de movimento**

Conforme visto existem diversas maneiras de conseguir capturar o movimento de um corpo. E existem diversas ferramentas que se aproveitaram de tais sistemas, é o caso do *Microsoft Kinect Sensor*, que faz uso de um sistema infravermelho e de câmeras *RGB*.

### **Kinect**

Desenvolvido pela *Microsoft*, o *Kinect* é um acessório para o console *Xbox* que possui a finalidade de capturar os movimentos feitos pelo jogador e compará-los com o do jogo ou

enviá-lo para que o jogo execute os movimentos. E para tal ele faz uso de uma câmera de profundidade *3D*, uma câmera *RGB*, microfones espalhados pela sua extensão e uma base com inclinação mecanizada (Xbox, 2013; HAN, 2013, tradução nossa), como visto na figura 4. As especificações e detalhes de cada componente podem ser encontrados no site oficial do Xbox e nos artigos de Han (2013), Strazdins, Komandur e Styve (2013) e Rocha, Defavari e Brandão (2013).

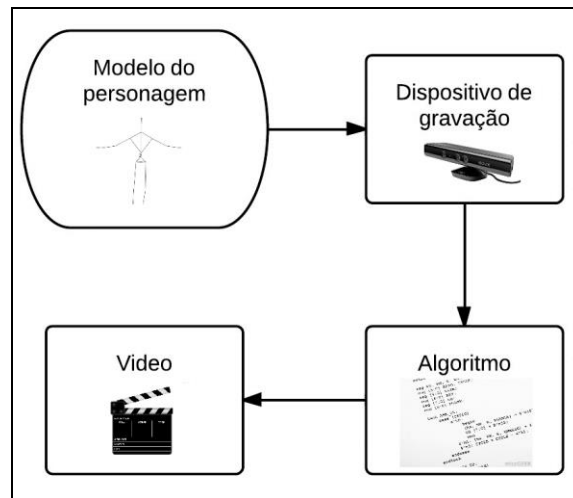


**Figura 17 – Componentes do Microsoft Kinect Sensor**

Tais componentes e funcionalidades, porém, somente eram aproveitadas pelo console *Xbox*, não sendo possível sua aplicação em outras áreas até o lançamento do *Framework Microsoft Kinect SDK*. Que, por sua vez, possui a finalidade de facilitar a utilização do Kinect por desenvolvedores.

### **Modelo de geração automática**

A criação de um modelo é sempre recomendada para auxiliar no entendimento da ferramenta que se pretende desenvolver. Com ele é possível identificar possíveis falhas ou aprimoramentos que necessitaram ser feitos antes do seu início, assim como fica mais fácil para quem não está diretamente envolvido com o projeto compreendê-lo. A figura 5 mostra como ficou a definição do modelo, a explicação de cada quadro pode ser verificada em seguida.



**Figura 18 - Modelo para geração automática de animação através da captura de movimento através do Kinect**

A primeira etapa consiste na elaboração do modelo do personagem a ser gerado pela captura, sendo decidido escolher uma forma genérica, apenas para fins de mostrar a movimentação dos membros no protótipo. Assim foi escolhido um modelo que ilustrasse as partes do corpo, interligadas adequadamente para tornar os movimentos os mais reais possíveis.

Porem para a obtenção de tal modelo, é necessário conseguir um dispositivo que ofereça a funcionalidade para captura de movimento. Sendo que existem diversos que oferecem tal recurso, sendo o Kinect o mais popular na atualidade.

Em conjunto com o dispositivo, é necessária a criação de alguma comunicação entre ele e o algoritmo, que torne possível o recebimento dos dados capturados e seu tratamento para a geração do modelo anterior. Sendo que essa geração precisa acontecer em tempo real e salva em algum formato que futuramente pode ser transformado em vídeo.

A transformação em vídeo pode ser feita de diversas maneiras, sendo esta a última etapa da modelagem. A forma como este vídeo será obtido é irrelevante, sendo apenas necessário que seja uma forma automática.

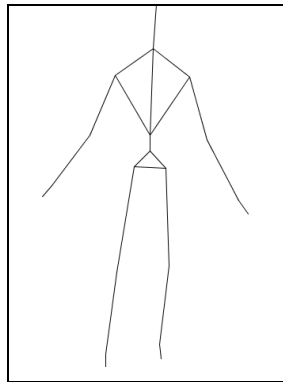
## Resultados

Dentre as três ferramentas analisadas, somente as experiências feitas com a última alcançaram o objetivo deste trabalho. As outras duas apresentaram problemas ou dificuldades que acabaram por atrasar o desenvolvimento do protótipo, motivo que levou a substituição das mesmas.

Com a utilização do *plug-in jnect* infelizmente não foi possível fazer conexão com o *Kinect* durante os testes. Após procurar uma solução para o problema foi verificado que era um erro que realmente estava acontecendo em algumas versões do *plug-in*, principalmente na última, porém o *download* das versões anteriores e da solução para o problema não estavam mais disponíveis. Fato que levou a sua troca pela biblioteca *SimpleOpenNI* e o ambiente *Processing*.

Com eles, a dificuldade foi transformar as imagens capturadas em um arquivo de vídeo. A biblioteca possui uma funcionalidade para gerar um arquivo “vídeo” no formato “oni”, formato lido somente através da utilização do *framework OpenNI*. No entanto não foi possível localizar um meio de transformar esse arquivo em um do tipo vídeo, para ser executado em um player comum no computador. Foram encontradas algumas soluções para trabalhar com os frames, mas não foram encontrados códigos para certificar de que esse método funcionaria.

A biblioteca *PKinect*, no entanto, se mostrou ideal para o cumprimento do objetivo do protótipo. Desde seus primeiros resultados, utilizando os códigos disponibilizados no site de seu criador, ela já mostrou que possuía todos os recursos necessários para, ao menos, conseguir mostrar e capturar os frames recebidos através do *Kinect*. A Figura 136 mostra outro código disponibilizado pelo criador da biblioteca em seu *website*.



**Figura 19 – Captura obtida com a utilização do protótipo**

A simplicidade e quantidade de material encontrado com a aplicação do *PKinect* reduziram consideravelmente a complexidade no desenvolvimento deste protótipo. Restando somente a conversão das imagens em um arquivo de vídeo, que foi alcançada satisfatoriamente através da aplicação *ffmpeg*.

Os testes feitos com o algoritmo foram muito satisfatórios. Em ambos os passos escolhidos foi possível a identificação dos membros de maneira fácil. No entanto um detalhe chamou a atenção durante os testes. Aparentemente o modelo esquelético pode apresentar alguns problemas caso o usuário faça uso de um cruzamento de membros. Em diversas situações os sensores confundiam os lados quando os braços ou pernas eram cruzados. Fato que acabou por colocar um pequeno limite nos passos que podem ser capturados pelo protótipo.

## **Análise**

O motivo principal para desenvolvimento deste protótipo foi fornecer uma base para que futuros pesquisadores pudessem vir a utilizar essa tecnologia com a finalidade de aplicá-las a outras áreas que poderiam vir a ser beneficiadas. Sendo o pequeno número de resultados encontrados durante as buscas, para utilização de tal tecnologia, outro fator de extrema relevância que levou a criação desta ferramenta.

Devido ao objetivo deste trabalho ser a criação de um protótipo, há diversas funcionalidades ou modificações que podem ser feitas, como por exemplo, a substituição do

modelo esquelético bidimensional por um tridimensional. Substituindo a imagem mostrada de um esqueleto para um personagem mais parecido com um ser humano. Tornando, assim, o algoritmo mais palpável para o ensino de passos ou rotinas de dança. Outra importante funcionalidade considerável seria a de verificar se os movimentos e posturas durante a coreografia estão corretos, ou se não estão forçando demais os músculos. O que pode ser útil se utilizado em um *software* que será repassado ao público, suprimindo parcialmente a necessidade de um profissional capacitado para as atividades físicas oferecidas pelo *software*. Mas este protótipo não está limitado à dança.

Na área da saúde ele pode ser amplamente aproveitado por desportistas, técnicos ou preparadores físicos. Com a obtenção do modelo esquelético poderia surgir a possibilidade de melhorar a performance dos atletas, como postura, inclinação, talvez até a força empregada em tal movimento.

A área de jogos pode ser igualmente favorecida com a utilização e complementação do protótipo. A criação de sequências de movimentos ficaria extremamente mais precisa e muito próxima da realidade. Tornando, talvez, o jogo mais facilmente aceito mesmo para os jogadores mais exigentes. Movimentos limpos, precisos e reais, eis um exemplo do que poderá ser obtido com alguns avanços. Mas não pode ser esquecida a área de animação, que poderá se aproveitar totalmente do algoritmo.

## **Conclusão**

Juntando e analisando todos os resultados encontrados pode-se confirmar que os objetivos deste trabalho foram alcançados. Podendo através da utilização de um Kinect a obtenção de uma animação do modelo esquelético do usuário em um formato de vídeo. Sendo ainda que tal protótipo pode ser amplamente melhorado.

*Upgrades* como a substituição dos traços do modelo esquelético por imagens respectivas da parte do corpo ou a inserção da opção de acoplamento de música ao vídeo, já melhorariam consideravelmente a qualidade e eficiência deste protótipo. Sendo ainda melhor a adaptação do mesmo para outras áreas, uma vez que tecnologias semelhantes estão sendo utilizadas nas áreas da saúde e esportes, poderia ser produtivo e dar início a um projeto ambicioso para melhoria na qualidade de vida.

## **Referencias**

CHUNG, B. (2013) “Magic & Love Interactive”, <http://www.magicandlove.com/blog>, Novembro

GOMIDE, J. V. B. et al (2013) “Captura de movimento e animação de personagens de jogo”, <http://lplace.dcc.ufmg.br/npdi/uploads/bd254485-3879-77ef.pdf>, Novembro

HAN, J. et al (2013) “Enhanced Computer Vision with Microsoft Kinect Sensor: A Review”, [http://lshao.staff.shef.ac.uk/pub/KinectReview\\_TC2013.pdf](http://lshao.staff.shef.ac.uk/pub/KinectReview_TC2013.pdf), Novembro

- MOREIRA, M. M. da S. (2013) “Qualidade de Vida: Expressões Subjetivas e Histórico-Sociais”, [http://www.uel.br/revistas/ssrevista/c-v9n1\\_marilda.htm](http://www.uel.br/revistas/ssrevista/c-v9n1_marilda.htm), Agosto
- PREVIDÊNCIA USIMINAS (2013) “Educação Financeira e Previdenciária: Artigos relacionados”, [http://www.previdenciausiminas.com/educacao\\_financeira\\_previdenciaria/artigos\\_relacionados/lazer\\_diversao\\_terceira\\_idade.php](http://www.previdenciausiminas.com/educacao_financeira_previdenciaria/artigos_relacionados/lazer_diversao_terceira_idade.php), Agosto
- PULLEN, K. A. (2013) “ Motion capture assisted animation: texturing and synthesis”, <http://nccastaff.bournemouth.ac.uk/hncharif/MathsCGs/Desktop/Research/General%20Papers/thesisD11.pdf>, Novembro
- ROCHA, P. R., DEFAVARI, A. H. e BRANDÃO, P. S. (2013) “Estudo da viabilidade da utilização do Kinect como ferramenta no atendimento fisioterapêutico de pacientes neurológicos”, <http://sbgames.org/sbgames2012/proceedings/papers/gamesforchange/g4c-04.pdf>, Novembro
- SALES, S. H. C. de (2013) “Animação de uma Marionete Virtual, a partir de um Sistema Óptico de Captura de Movimento Humano, sem Utilização de Marcação Especiais”, <http://campeche.inf.furb.br/tccs/2004-II/2004-2silviahcsalesvf.pdf>, Novembro
- SALVE, M. G. C. (2013) “A prática da atividade física: Estudo comparativo entre os alunos de graduação da UNICAMP”, <http://www.scielo.gpeari.mctes.pt/pdf/mot/v4n3/v4n3a06.pdf>, Agosto
- SISTO, T. M. e CABRAL, E. L. (2013) “Sistema de captura de movimentos com sensores infravermelho”, <http://www.maua.br/pesquisas/sistema-de-captura-de-movimento-com-sensores-infravermelho>, Novembro
- Strazdins, G., Komandur, S. e Styve, A. (2013) “Kinect-based systems for maritime operation simulators?”, [http://www.scs-europe.net/dlib/2013/ecms13papers/ese\\_ECMS2013\\_0041.pdf](http://www.scs-europe.net/dlib/2013/ecms13papers/ese_ECMS2013_0041.pdf) , Dezembro
- Xbox (2013) “Componentes do Sensor Kinect”, <http://support.xbox.com/pt-BR/xbox-360/kinect/kinect-sensor-components>, Novembro