



CENTRO UNIVERSITÁRIO LUTERANO DE PALMAS

COMUNIDADE EVANGÉLICA LUTERANA "SÃO PAULO"
Recredenciado pela Portaria Ministerial nº 3.607 - D.O.U. nº 202 de 20/10/2005

Gleyson Rocha Moura

UTILIZANDO O MICROSOFT KINECT NA OBTENÇÃO DE ATRIBUTOS
ANTROPOMÉTRICOS

Palmas - TO

2014

Gleyson Rocha Moura
UTILIZANDO O MICROSOFT KINECT NA OBTENÇÃO DE ATRIBUTOS
ANTROPOMÉTRICOS

Trabalho de Conclusão de Curso (TCC) elaborado e apresentado como requisito parcial para obtenção do título de bacharel em Sistemas de Informação pelo Centro Universitário Luterano de Palmas (CEULP/ULBRA).

Orientador: Prof. M.Sc. Fernando Luiz de Oliveira.

Palmas - TO

2014

Gleyson Rocha Moura
UTILIZANDO O MICROSOFT KINECT NA OBTENÇÃO DE ATRIBUTOS
ANTROPOMÉTRICOS

Trabalho de Conclusão de Curso (TCC) elaborado e apresentado como requisito parcial para obtenção do título de bacharel em Sistemas de Informação pelo Centro Universitário Luterano de Palmas (CEULP/ULBRA).

Orientador: Prof. M.Sc. Fernando Luiz de Oliveira.

Aprovada em 26 de novembro de 2014.

BANCA EXAMINADORA

Prof. M.Sc. Fernando Luiz de Oliveira
Centro Universitário Luterano de Palmas

Prof. M.Sc. Fabiano Fagundes
Centro Universitário Luterano de Palmas

Prof. M.Sc. Pierre Soares Brandão
Centro Universitário Luterano de Palmas

Palmas - TO

2014

Dedico este trabalho primeiramente a Deus pelo amor incondicional e aos meus pais, Maria Cleudia e Sebastião Mecnas que pelo exemplo de vida, carinho e compreensão, incentivaram-me ao longo da trajetória acadêmica.

AGRADECIMENTOS

Agradeço primeiramente a Deus, que foi o responsável por tudo existir e pela capacidade me dada para suportar todas as dificuldades. Foi Ele quem me guiou durante todos os dias de luta desde os primeiros dias de aula, que não foram nada fáceis, foi Ele o responsável pela minha coragem em continuar durante os momentos de desespero, onde o cansaço era maior do que a força de vontade. Ele quem sempre me mostrou o melhor caminho a seguir, iluminando os meus passos e me ajudando a cumprir as missões me oferecidas.

Agradeço aos meus familiares, que sempre me apoiaram e que sempre foram os maiores incentivadores desta vitória. Primeiro aos meus pais que sempre me ajudaram todas as dificuldades. Agradeço a Deus pela chance de ser filho dos melhores pais desse mundo. Segundo as minhas queridas irmãs (Suâny e Jakeline) por me ajudar e dar apoio em todos os momentos da minha vida, a minha linda e preciosa namorada (Bárbara) por toda paciência e incentivos durante essa jornada e principalmente pelas “ajudinhas” nas correções do TCC. Enfim agradeço a todos da minha família.

Agradeço também ao Fernando Luiz de Oliveira meu orientador em todos os meus trabalhos científicos, pelas considerações, orientações e pela paciência, enfim, tenho certeza que teve papel fundamental na minha formação. Muito obrigado Fernando.

Agradeço todos os professores do curso em especial ao professor Edeilson, por todas as conversas, conselhos e orientação e ao Fabiano, que nos momentos difíceis sempre me apoiou, incentivou e direcionou-me para chegar a este momento. Obrigado “gordinho”.

Agradeço ao meu brother Carlos Diego, meu fiel companheiro/amigo, não apenas na faculdade, por toda amizade, por compartilharmos vários momentos difíceis que enfrentamos e por todas as vitórias que foram conquistadas em conjunto. Também aos meus colegas de turma.

Enfim, agradeço a todos, que me ajudaram diretamente ou indiretamente no desenvolvimento do curso. Foram anos de lutas e glórias. Isto é um sonho realizado que jamais esquecerei. Muito obrigado.

RESUMO

O trabalho trata do desenvolvimento de uma aplicação que utiliza o Sensor Kinect para obter um conjunto de atributos antropométricos. Para proporcionar melhorias no que tange a área da fisioterapia, uso de técnicas computacionais têm sido cada vez mais exploradas. Neste sentido o Microsoft Kinect, possibilita a captura e mapear a superfície do corpo humano e consequentemente informações relevantes sobre o mesmo. Neste contexto, pesquisas científicas, acerca do sensor Kinect, têm surgido a fim de contribuir significativamente com os procedimentos fisioterapêuticos, possibilitando desde a recuperação física até a obtenção das características físicas de pacientes. Assim este projeto visa apresentar uma aplicação computacional que consiga capturar e analisar os atributos antropométricos de certo indivíduo, de forma automática. Para isso, será necessário compreender o conceito da antropometria, que se relaciona ao estudo das medidas do corpo humano, e consequentemente, como obter características físicas utilizando o sensor Microsoft Kinect.

Palavras-chave: Kinect, Antropometria, Fisioterapia.

LISTA DE FIGURAS

Figura 1 - Representação triortogonal (SIRGADO, 2005, p. 6).....	14
Figura 2 - Localização dos pontos anatômicos (BRENDLER, 2013, p. 70).....	15
Figura 3 - Exemplo de dados antropométricos estáticas (IIDA 2005, p.117).	16
Figura 4 - Medidas antropométricas dinâmicas (IIDA, 2005, p.123).....	17
Figura 5 - Estatura corpo ereto, adaptado de (IIDA 2005, p.117).....	18
Figura 6 - Largura dos ombros, em pé, adaptado de (IIDA 2005, p.117).	19
Figura 7 - Altura da Tíbia como o corpo ereto, adaptado de (IIDA 2005, p.117).....	19
Figura 8 - Largura dos quadris, corpo ereto adaptado de (IIDA 2005, p.117).....	20
Figura 9 - Comprimento do membro superior, adaptado de (IIDA 2005, p.117).	20
Figura 10 - Gráfico da distribuição normal padrão (ZELNIK, PANERO, 2001 p. 33).	24
Figura 11 - Distribuição cumulativa da estatura (ZELNIK, PANERO, 2001 p. 33).....	25
Figura 12 - Hierarquia de tarefas computacionais (MOLZ, 2001 p. 18).....	27
Figura 13 - Tópicos da visão computacional (JUNGONG et al. 2013, p. 1319).	28
Figura 14 - Referência Microsoft.Kinect.....	31
Figura 15 - Arquitetura do funcionamento da aplicação.	33
Figura 16 - Método IniciarSensorKinect().	34
Figura 17 - Diagrama de Classe, classes ColorImageFrame e ColorImageFrame (MSDN, 2014, online).....	35
Figura 18 - Método CameraImagemRGB().....	36
Figura 19 - Método Kinect_TodosQuadros().	37
Figura 20 - Interface com Imagem RGB.	37
Figura 21 - Campo de visão dos dados de profundidade do Kinect (JANA, 2012, p. 11).	39
Figura 22 - Método IniciarSensorKinect().	40
Figura 23 - Câmera de profundidade.	40
Figura 24 - Eixos do esqueleto adaptado (CATUCHE, 2012, p. 21).	41
Figura 25 - Diagrama de classe, classe Skeleton (MSDN, 2014, online).....	42
Figura 26 - Método FuncoesEsqueletoUsuario().	43
Figura 27 - Conjunto de articulação sensor Kinect (JANA, 2012, p. 163).	44
Figura 28 - Extensão do método FuncoesEsqueletoUsuario().	44
Figura 30 - Método DistanciaEntrePontos().....	45
Figura 31 - Interface da aplicação de resultados	46

LISTA DE TABELAS

Tabela 1 - Tamanho da Amostra Roebuck (1975 apud BOUERI, 2008, p. 59).....	23
Tabela 2 - Coeficiente para Cálculo do Percentil (BOUERI, 2008, p. 75).	25
Tabela 3 - Resoluções e formatos propriedade ColorImageFormat (WEBB; ASHLEY, 2001).	36

LISTA DE EQUAÇÕES

Equação 1 - Cálculo da média aritmética (RON, BETSY, 2008 p. 159).	21
Equação 2 - Cálculo do Desvio padrão (BASTOS, DUQUIA, 2007, p. 42).....	22
Equação 3 - Cálculo da dimensão da amostra (MORRE, 2000, p.168).	22
Equação 4 - Cálculo da dimensão precisa da amostra (MORRE, 2000, p.168).	23
Equação 5 - Teorema de Pitágoras.	45

LISTA DE ABREVIATURAS

CEULP - Centro Universitário Luterano de Palmas – TO

IDE - Integrated Development Environment

IR - Infra Red (infravermelho)

RGB - Padrão de cores para imagens (Vermelho, Verde, Azul)

SDK - Software Development Kit

WPF - Windows Presentation Foundation

XAML - Extensible Application Markup Language

SUMÁRIO

1 INTRODUÇÃO.....	12
2 REFERENCIAL TEÓRICO.....	14
2.1 Pontos Anatômicos	14
2.2 Antropometria.....	16
2.2.1 <i>Medidas antropométricas estáticas</i>	18
2.3 Cálculos estatísticos.....	21
2.3.1 <i>Amostra</i>	21
2.3.2 <i>Percentis</i>	24
2.4 Visão Computacional.....	26
3 MATERIAS E MÉTODOS	30
3.1 Local e Período	30
3.2 Matérias	30
3.3 Metodologia.....	31
4 RESULTADOS E DISCUSSÃO.....	33
4.1 Visão geral da aplicação	33
4.2 Fluxo de Cores de Imagem.....	34
4.3 Fluxo de Profundidade.....	38
4.4 Fluxo de esqueleto	41
5 CONSIDERAÇÕES FINAIS	49
6 REFERÊNCIAS	51

1 INTRODUÇÃO

No campo tecnológico, atualmente, há um crescente interesse na utilização de técnicas computacionais, com intuito de proporcionar melhorias nas diversas áreas do conhecimento. No que tange à fisioterapia, têm surgido inúmeras pesquisas científicas que buscam o desenvolvimento de aplicações para auxiliar nos procedimentos fisioterapêuticos, desde a recuperação física até a obtenção de características físicas.

Pelegri (2011) afirma que algumas destas pesquisas envolvem a antropometria, que consiste em ser um método de investigação científica cujo objetivo visa à medição das dimensões físicas e das suas variações que compõem o corpo humano.

Os atributos antropométricos são amplamente utilizados na área da saúde, sendo de grande relevância na avaliação e acompanhamento de diversos aspectos relacionados a saúde humana (ROEBUCK et al., 1975, p. 18). Ou seja, a antropometria é definida como o modo de determinar objetivamente as medidas do corpo humano, permitindo evidenciar a análise quantitativa das medidas como o tamanho, a forma e a composição corporal do ser humano.

Segundo Donati et al., (2008, p. 2220) através do esqueleto humano é possível identificar os pontos de anatômicos. Estes, são pontos esqueléticos que são identificados próximos à superfície do corpo através da palpação. Sendo assim possível de determinar os locais de medição.

Os atributos antropométricos são obtidos através de dois métodos, diretos e indiretos. No método direto há o contato físico com o indivíduo a ser mensurado e envolvem o uso de instrumentos de medição. Já no método indireto, para aquisição de grande quantidade de variáveis do corpo humano são utilizados recursos como digitalizadores bidimensionais ou tridimensionais. Entretanto, a utilização destes métodos é limitado ao contato físico como o indivíduo a ser mensurado, o que torna o procedimento demorado.

Com isso, o presente trabalho apresenta uma aplicação que obtém um conjunto de atributos antropométricos de um paciente, utilizando o Sensor Microsoft Kinect. Este, por sua vez, é um dispositivo que possui um projetor infravermelho, base em laser e duas câmeras, sendo uma capaz de representar objetos em profundidade e outra câmera de vídeo colorida, que captura cores, como vermelho, verde e azul (RGB). Os dados capturados pelo sensor fornecem detalhes suficientes sobre as características físicas de um determinado indivíduo.

Apesar da complexidade envolvida na captura de atributos antropométricos, o sensor Kinect auxilia na captura e pré-processamento de imagens. Assim é possível a sistematização do processo de captura de atributos antropométricos, por meio de seu sensor tridimensional, permitindo aos fisioterapeutas o acesso das medidas de um indivíduo a partir de uma visão real.

Este trabalho está organizado em seções, sendo que a seção 2 refere-se à revisão de literatura, contendo os principais conceitos envolvidos no trabalho; em seguida, seção 3, são apresentados os materiais e métodos, contemplando tanto a parte de construção do trabalho como o planejamento de sua execução. Os resultados alcançados são apresentados na seção 4; Na seção 5 são apresentadas as considerações finais obtidas ao termino do trabalho e, por fim, na seção 6 as referências bibliográficas utilizadas.

2 REFERENCIAL TEÓRICO

Nesta seção serão abordados os conceitos relevantes e necessários para compreensão do trabalho, como: Pontos Anatômicos, Antropometria, cálculos estatísticos e a visual computacional.

2.1 Pontos Anatômicos

Através do esqueleto humano é possível identificar os pontos de anatômicos. Estes são pontos esqueléticos que são identificados próximos à superfície do corpo através da palpação (DONATI, et al., 2008, p. 2220). Por meio destes pontos, é possível determinar os locais de medição. Assim, ao realizar processo de identificação dos pontos anatômicos, o indivíduo deve se encontrar na posição anatômica, isto é, “na posição de corpo vivo, em pé, com braços ao longo do corpo e as palmas das mãos viradas para a frente” (BASMAJIAN, 1985, *apud* BRENDLER, 2013, p. 68).

Desse modo, Norton e Olds (2005, p. 67) afirmam que, para realizar o processo de medição dos atributos antropométricos é essencial conhecer a anatomia humana para assim identificar os locais dos pontos anatômicos, bem como permitir que se adquira a padronização das medidas. Estes mesmos autores explicam que o corpo humano pode ser dividido em três planos principais, conforme apresentado na figura 1.

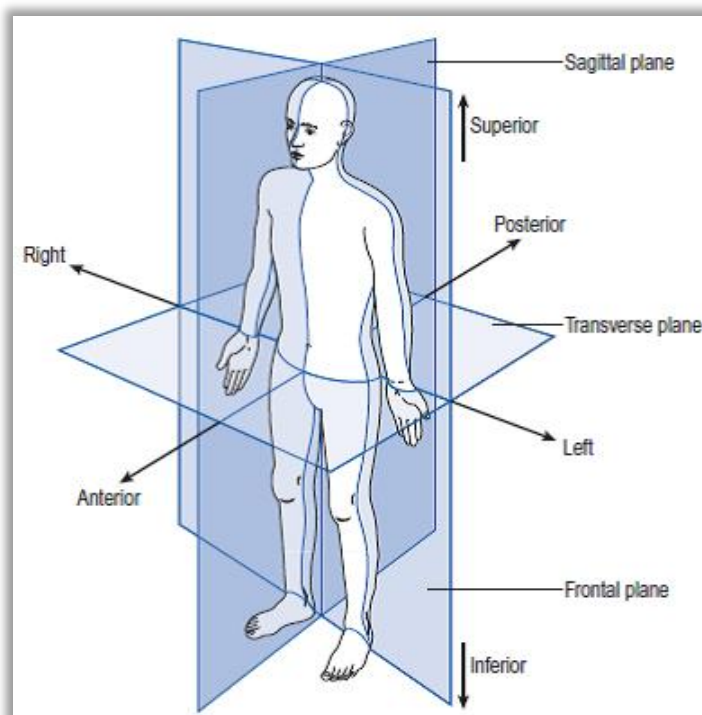


Figura 1 - Representação triortogonal (SIRGADO, 2005, p. 6).

A figura 1 apresenta a representação do corpo humano na forma triortogonal, ou seja, a divisão do corpo em três planos. Os planos são responsáveis por dividir (cortar) o corpo humano em posição anatômica. Para facilitar o entendimento de cada plano, a seguir, os mesmos serão detalhados, conforme entendimento dos autores citado anteriormente:

- Plano sagital: divide o corpo nos planos sagital direita e sagital esquerda. As ações articulares ocorrem em torno de um eixo horizontal ou transversal;
- Plano coronal ou frontal: divide o corpo em partes anterior (ventral) e posterior (dorsal). As ações articulares ocorrem em torno de um eixo ântero-posterior;
- Plano transversal ou horizontal: divide o corpo em partes superior (cranial) e inferior (caudal). As ações articulares ocorrem em torno do eixo longitudinal ou vertical.

A figura 2, apresentada a seguir, exemplifica melhor a posição do corpo humano na posição anatômica, bem como a identificação dos pontos de referência.

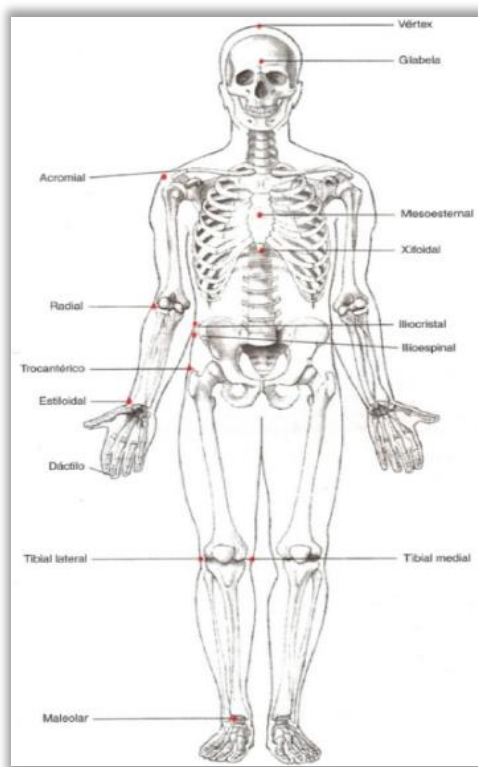


Figura 2 - Localização dos pontos anatômicos (BRENDLER, 2013, p. 70).

Nota-se que na figura 2 alguns pontos anatômicos nomeados e identificados com pontos vermelhos. Estes estão assim identificados por serem obtidos por meio de palpação, representando uma pequena quantidade de locais de referência possíveis de se encontrar no

corpo humano, sendo considerados pontos anatômicos básicos para uma padronização na Antropometria (NORTON; OLDS, 2005, p. 52).

É importante ressaltar que mesmo obtendo o conhecimento da localização dos pontos anatômicos, pode-se haver inconstância nas medidas antropométricas, mesmo refazendo os procedimentos de obtenção das medidas. Esta inconstância decorre da diversidade biológica ou em razão das variações de execução das técnicas de medição.

Após conhecer a anatomia básica do corpo humana e a identificação dos pontos anatômicos para obtenção das medidas antropométricas, pode-se seguir para a próxima seção, que tratará da Antropometria.

2.2 Antropometria

De acordo com Boueri (1991, p. 21), a antropometria é a ciência que estuda as medidas do corpo humano para determinar os aspectos referente às variações humanas. Ou seja, a antropometria é definida como o modo de se determinar objetivamente as medidas do corpo humano, permitindo evidenciar a análise quantitativa das medidas como tamanho, a forma e a composição corporal do ser humano.

As medidas antropométricas são amplamente utilizadas na área da saúde, sendo de grande relevância na avaliação e acompanhamento de diversos aspectos relacionados a saúde humana (ROEBUCK et al., 1975, p. 18).

A antropometria pode ser dividida em estática e dinâmica. A estática diz respeito às dimensões estruturais do corpo humano, na posição parada ou com poucos movimentos. As medições são realizadas entre os pontos anatômicos identificados. A figura 3, apresentada a seguir, elucida melhor os dados antropométricos estáticos.

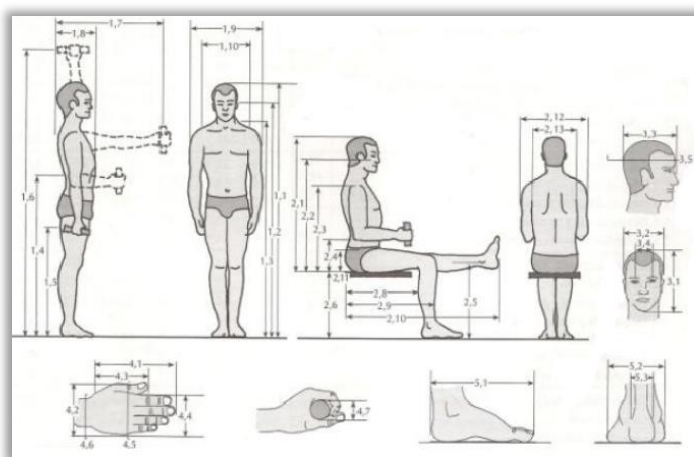


Figura 3 - Exemplo de dados antropométricos estáticas (IIDA 2005, p.117).

A antropometria dinâmica, segundo Iida (2005, p. 110), corresponde às medidas referentes ao alcance de uma extensão de movimento de uma articulação ou da força das várias ações da articulação. Isto é, são medidas de cada movimento realizado pelo corpo relacionado a alguma tarefa específica mantendo o restante do corpo de forma estática. Na figura 4 são apresentadas as medidas referentes à antropometria dinâmica.

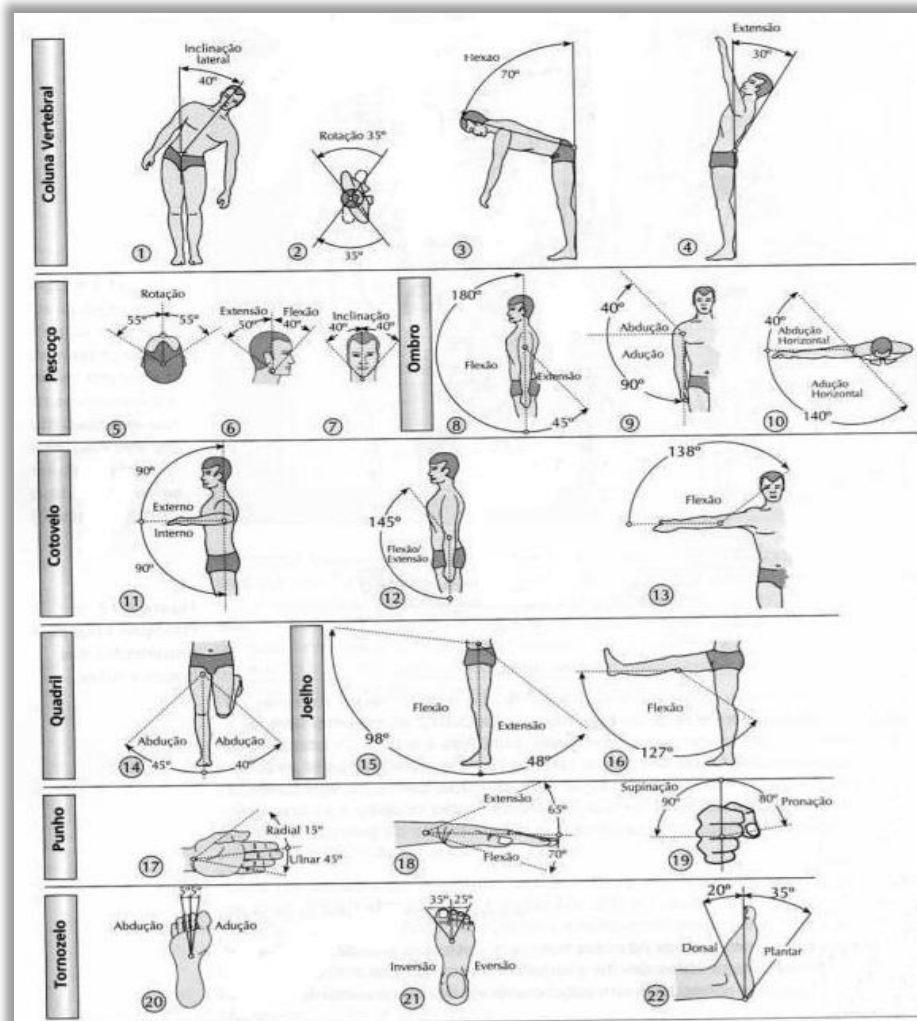


Figura 4 - Medidas antropométricas dinâmicas (IIDA, 2005, p.123).

De acordo com Brendler (2013, p. 44), as técnicas para realizar as medições tanto na antropometria estática quanto na antropometria dinâmica classificam-se basicamente em dois métodos:

- Métodos diretos ou de contato: envolvem o uso de instrumentos de medição que entram em contato físico com o indivíduo. Tais instrumentos são: fita métrica, antropômetro,

paquímetro, entre outros. Iida (2005, p. 110) enfatiza que, cada medição a ser efetuada deve especificar três fatores: localização, direção e postura. A localização indica o ponto do corpo a ser medido a partir de outro de referência. A direção indica se o comprimento será medido na horizontal ou vertical. Já a postura indica a posição do corpo, sentado, em pé ereto ou relaxado. Em diversas medições, o indivíduo é colocado na postura ereta, com os segmentos corporais nos ângulos de 180, 0, ou 90 graus em relação aos pontos demarcados (BRENDLER, 2013, p. 44). Esta postura é chamada de Posição Anatômica conforme citado na seção 2.1.

- Métodos indiretos: trata-se do uso de recursos como digitalizadores bidimensionais ou tridimensionais na aquisição de grande quantidade de variáveis do corpo humano, sendo utilizado especificamente na antropometria dinâmica. Dentre vários recursos disponíveis, a fotogrametria digital é a que se mais se destaca (BOUERI, 2008, p. 51), sendo uma técnica que representa as características geométricas dos corpos humanos a partir de imagens fotográficas bidimensionais.

Cabe ressaltar que este projeto delimita-se apenas à obtenção das variáveis da antropometria estática, mais especificamente na obtenção da estatura, largura dos ombros, altura da tíbia, largura dos quadris e comprimento do membro superior. O processo de captura dessas variáveis será abordado na próxima subseção.

2.2.1 Medidas antropométricas estáticas

De acordo com Petroski (2007, p. 09-31), as medidas antropométricas estáticas são obtidas da seguinte forma:

- Para medição antropométrica da estatura com o corpo ereto, o indivíduo a ser mensurado deve estar na posição ortostática, com os pés descalços, unidos e encostado na parede. Geralmente neste método utiliza-se um equipamento apropriado denominado estadiômetro.

Figura 5.

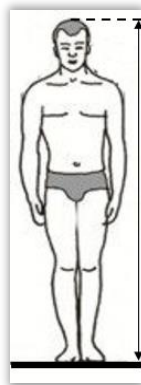


Figura 5 - Estatura corpo ereto, adaptado de (IIDA 2005, p.117).

- A medição da largura dos ombros com o corpo ereto consiste na distância entre as articulações dos ombros do indivíduo, sendo usado o equipamento paquímetro, direcionado para cima, conforme figura 6.

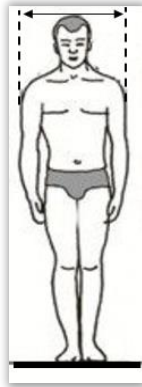


Figura 6 - Largura dos ombros, em pé, adaptado de (IIDA 2005, p.117).

- Para a medição da altura da Tíbia, com o corpo ereto, o indivíduo deve ficar na posição ortostática, com os pés descalços e unidos. Depois posiciona-se a haste da fita métrica metálica no ponto anatômico protuberância da tíbia conduzindo-a até o chão, como ilustra a figura 7.



Figura 7 - Altura da Tíbia como o corpo ereto, adaptado de (IIDA 2005, p.117).

- Para a medição da largura dos quadris, com o corpo ereto, têm-se como referência o ponto anatômico denominado crista ilíaca. Neste caso, utiliza-se o equipamento paquímetro, como demonstra a figura 8.



Figura 8 - Largura dos quadris, corpo ereto adaptado de (IIDA 2005, p.117).

- Para o comprimento do membro superior, na horizontal até o centro da mão, o indivíduo se posiciona com corpo ereto e com o braço direito estendido na horizontal na altura dos ombros, utilizando uma fita métrica para a obtenção da medida. O ponto de medição vai da articulação do ombro até o final do dedo médio, conforme figura 9.

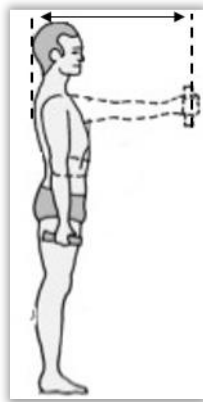


Figura 9 - Comprimento do membro superior, adaptado de (IIDA 2005, p.117).

As medidas antropométricas obtidas, independentemente do método utilizado, necessitam de uma organização de forma que seja possível apresentar dados na forma de estatísticas. A organização da pesquisa antropométrica e estudos dos cálculos estatísticos serão tema da próxima seção.

2.3 Cálculos estatísticos

A estatística é definida como a ciência que indica uma metodologia que coleta, classifica, apresenta, analisa e interpreta dados quantitativos e utiliza esses dados para a tomada de decisão (RON BETSY, 2008 p. 3). Os métodos estatísticos auxiliam o pesquisador a obter conclusões sobre um determinado problema a partir de um conjunto de informações do referido problema.

Na antropometria é de suma importância a utilização de cálculos estatísticos, uma vez que, organizar estatisticamente os dados antropométricos, permite que esses dados sejam facilmente obtidos e interpretados. Nesse sentido Boueri (2008, p.59) afirma que, primeiramente deve-se ter os dados referentes a amostra das medidas antropométricas, e ordená-los de um modo lógico, sob o ponto de vista estatístico, sendo, portanto, discutido na sequência deste trabalho.

2.3.1 Amostra

Estudos relacionados a elementos da população permitem o conhecimento de suas variáveis. Entretanto, nem sempre é possível obter as informações de todos os elementos da população devido a abrangência de fatores. Por isto, justifica-se o uso de amostras (LEVIN et al., 2012, p.156). Assim, Toledo e Ovalle (2008, p.63) definem amostra como um subconjunto de uma parte selecionada da totalidade de observações abrangidas pela população, através da qual se faz inferência sobre as características da população.

A amostra tende a apresentar uma distribuição semelhante à da população. Para se ter uma representação da distribuição de uma certa amostra, utilizam-se os estimadores que são respectivamente, a média e o desvio-padrão.

- A média aritmética é um valor representativo da amostra de um conjunto de valores. Para calcular o valor da média, aplica-se a seguinte fórmula.

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n}$$

Equação 1 - Cálculo da média aritmética (RON, BETSY, 2008 p. 159).

- O desvio-padrão é uma medida de dispersão que estima o quanto, em média, cada valor se distancia da própria média aritmética de uma distribuição, com a vantagem de preservar a unidade de mensuração original das observações (BASTOS, DUQUIA, 2007, p. 42). A fórmula para obter o desvio-padrão é apresentada pela equação 2.

$$s = \sqrt{\frac{\sum (x - \bar{X})^2}{(n - 1)}}$$

Equação 2 - Cálculo do Desvio padrão (BASTOS, DUQUIA, 2007, p. 42).

Se diversas amostras aleatórias forem adquiridas de uma certa população, certamente influenciará no valor médio da população e conseqüentemente no desvio-padrão. Neste caso, a variabilidade das médias é estimada através do erro padrão (LUNET et al., 2006 p. 55).

Hume e Marfell-Jones (2008, p. 1334) afirmam que, uma medida é considerada confiável quando há pequenas variações na média, um baixo erro padrão de medição na média e uma alta correlação entre medições em testes.

De acordo com Lunet et al. (2006, p, 55)

Amplitudes de prováveis erros da amostragem são geralmente expressas em termos dos limites de confiança de 95% do parâmetro em causa, que são definidos por $\pm 1,96$ de erros-padrão, os verdadeiros valores de qualquer parâmetro da população estarão dentro de $\pm 1,96$ de erros-padrão da estatística, 95 vezes em cada 100 amostras que forem obtidas.

Nos estudos antropométricos, pode-se aplicar os limites de confiança de 95% (d), sendo possível através da equação 3.

$$d = \frac{K_1 \cdot s}{\sqrt{n}}$$

Equação 3 - Cálculo da dimensão da amostra (MORRE, 2000, p.168).

Onde, k_1 é a constante para cada estatística, conforme apresenta a tabela 1, s é o desvio-padrão, n é a dimensão necessária da amostra e d é a precisão desejada para a medição ($\pm d$ unidades). Sendo possível também a utilização da equação 4.

$$n = \left(\frac{K_1 \cdot S}{d} \right)^2$$

Equação 4 - Cálculo da dimensão precisa da amostra (MORRE, 2000, p.168).

Nota-se que na equação 4 pode-se indicar a quantidade de sujeitos que devem ser medidos para obter a estatística em causa com o grau de precisão aquedada. Esta precisão reduz-se a dimensão da amostra inicial N_i é inferior a 100. Sendo assim, pode-se obter uma estimativa mais precisa usando k_2 em vez do k_1 , conforme Roebuck exemplifica na tabela .

Tabela 1 - Tamanho da Amostra Roebuck (1975 apud BOUERI, 2008, p. 59).

Valores de k_1 para determinação do Tamanho da Amostra		
Interesse estatístico	k_1	
Media	1.96	Percentil
Desvio-padrão	1.39	Percentil
50°	2.46	Percentil
45° e 55°	2.46	Percentil
40° e 60°	2.49	Percentil
35° e 65°	2.52	Percentil
30° e 70°	2.58	Percentil
25° e 75°	2.6	Percentil
20° e 80°	2.80	Percentil
15° e 85°	3.00	Percentil
10° e 90°	3.35	Percentil
5° e 96°	4.14	Percentil
4° e 96°	4.46	Percentil
3° e 97°	4.92	Percentil
3° e 98°	5.67	Percentil
1° e 99°	7.33	Percentil
Obs.: quando $N < 100$ usar k_2 para $100 < N > 40$ $k_2 - 2,00$ para $40 < N > 20$ $k_2 - 2,03$ para $20 < N > 10$ $k_2 - 2,16$		

Nem sempre é possível compreender o significado dos dados antropométricos contidos em uma amostragem. Para isto, os dados antropométricos, devem ser ordenados de modo a indicar a frequência em que ocorrem. Zelnik e Panero (2001, p.32) dizem que, para melhor visualização dos dados antropométricos é necessário o uso de métodos da distribuição de frequência.

Os dados antropométricos são apresentados frequentemente através da distribuição normal ou Curva de Gauss (BOUERI, 2008, p. 62). Esta distribuição tem como objetivo realizar uma descrição da forma em que uma população é distribuída, isto é, conhecendo-se a forma, a

média e o desvio-padrão, pode-se ter a representação de uma população. A figura 10 ilustra o gráfico da distribuição normal padrão com os parâmetros média e desvio-padrão.

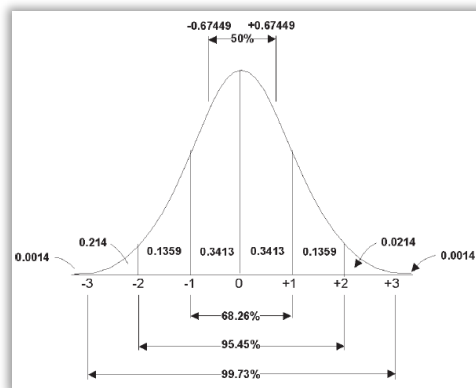


Figura 10 - Gráfico da distribuição normal padrão (ZELNIK, PANERO, 2001 p. 33).

Nota-se que é possível verificar que as percentagens de medições localizadas entre os intervalos são descritos pelos termos do desvio-padrão. Isto é, cerca de 95,45% das medições estão compreendidas no intervalo $[-2\sigma, 2\sigma]$ centrado na média. Como a curva é simétrica, 50% das medições são inferiores à média e 50% são-lhe superiores (MORAES).

2.3.2 Percentis

Na pratica a maior parte dos dados antropométricos são expressos e utilizados em percentis. Pequini (2015, p. 411) afirma que percentis são usados para entender bem uma distribuição. Para isso, se torna necessário ter o conhecimento dos valores acima ou abaixo dos quais se encontra uma determinada percentagem dos dados.

De acordo com Moraes (1983, p. 257)

Devido às significantes variações nos tamanhos do corpo de cada indivíduo, as médias são irrelevantes. Portanto é necessário trabalhar com limites de variação. Estatisticamente, pode-se constatar que as medidas do corpo humano, de qualquer população dada, distribuem-se de tal modo que, de alguma maneira, cairão em algum lugar pelo meio da curva de distribuição, enquanto um pequeno número de medidas extremas cairá em uma das extremidades do espectro. Por isso, a maioria dos dados antropométricos é, em geral, apresentada sob a forma de percentis.

A figura 11 exemplifica os dados antropométricos na forma de percentis.

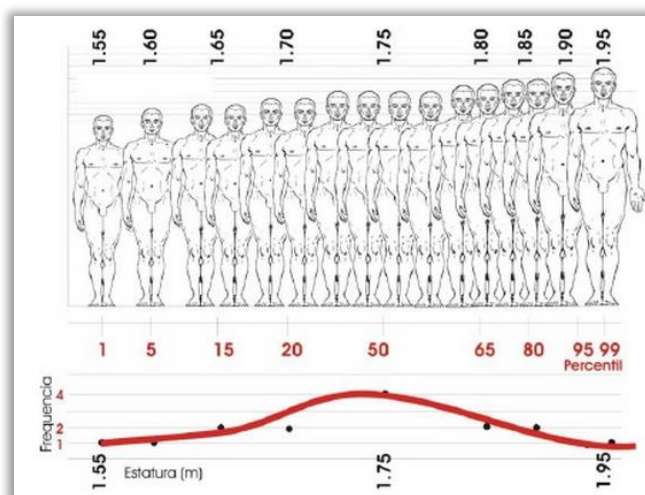


Figura 11 - Distribuição cumulativa da estatura (ZELNIK, PANERO, 2001 p. 33).

O cálculo para se ter os valores em percentis é realizado somando ou subtraindo a média da dimensão com o desvio-padrão, que por sua vez é multiplicado pelo fator k_1 que representa os valores percentis na curva de Gauss. A tabela 3 apresenta o coeficiente k_1 para o cálculo dos valores em percentis.

Tabela 2 - Coeficiente para Cálculo do Percentil (BOUERI, 2008, p. 75).

Coeficiente k_1 para Cálculo do Percentil		
Percentil	k_1	Fórmula para cálculo dos limites
0,5	2,58	$\bar{x} - (2,58 \cdot S)$
1	2,32	$\bar{x} - (2,32 \cdot S)$
2,5	1,95	$\bar{x} - (1,95 \cdot S)$
3	1,88	$\bar{x} - (1,88 \cdot S)$
5	1,65	$\bar{x} - (1,65 \cdot S)$
10	1,28	$\bar{x} - (1,28 \cdot S)$
20	0,80	$\bar{x} - (0,84 \cdot S)$
25	0,67	$\bar{x} - (0,67 \cdot S)$
30	0,52	$\bar{x} - (0,52 \cdot S)$
50		
70	0,52	$\bar{x} + (0,52 \cdot S)$
75	0,67	$\bar{x} + (0,67 \cdot S)$
80	0,80	$\bar{x} + (0,84 \cdot S)$
90	1,28	$\bar{x} + (1,28 \cdot S)$
95	1,65	$\bar{x} + (1,65 \cdot S)$
97	1,88	$\bar{x} + (1,88 \cdot S)$
97,5	1,95	$\bar{x} + (1,95 \cdot S)$
99	2,32	$\bar{x} + (2,32 \cdot S)$
99,5	2,58	$\bar{x} + (2,58 \cdot S)$

Nesta seção foram apresentados conceitos pertinentes sobre os cálculos estatísticos. Na próxima seção serão abordados os conceitos relacionados a visão computacional.

2.4 Visão Computacional

O sentido da visão pode ser considerado o meio mais eficiente que o ser humano dispõe para capturar as informações originadas no ambiente que o cerca (MOLZ, 2001 p. 17). A forma que a visão humana processa e interpreta grandes quantidades de informações motivou diversos estudos científicos, com o propósito de desenvolver técnicas e dispositivos que estenda sua capacidade de percepção visual. Nesse contexto, a visão computacional é a ciência que busca reproduzir através de técnicas as sensações e estímulos percebidos pelos órgãos de visão em um ambiente computacional (BARROS; PINHEIRO, 2013, p. 3527).

Entende-se que a visão computacional é definida como a ciência que se dedica ao desenvolvimento de técnicas que possibilitam a aquisição, processamento e entendimento de imagens para o computador. Estas imagens são obtidas por dispositivos, tais como sensores tridimensionais, câmeras, entre outros.

Marr (1982, p. 31) definiu a visão como “um processo que produz uma descrição, a partir de imagens do mundo externo, que é útil ao visualizador e não repleta de informações irrelevantes”. Na visão do autor, o processo realiza o mapeamento de diversos aspectos de uma cena, contida no ambiente externo, extraídas através de matrizes de valores de intensidade de brilho aos diferentes padrões que apresentam esta cena. Estas matrizes são adquiridas nos primeiros estágios da visão computacional, por outro lado os diferentes padrões são adquiridos nos últimos estágios da mesma (MOLZ, 2001 p. 18).

O processo de visão definido por Marr, é trivialmente implementado por uma sequência de diferentes tarefas, conforme apresenta a figura 12.

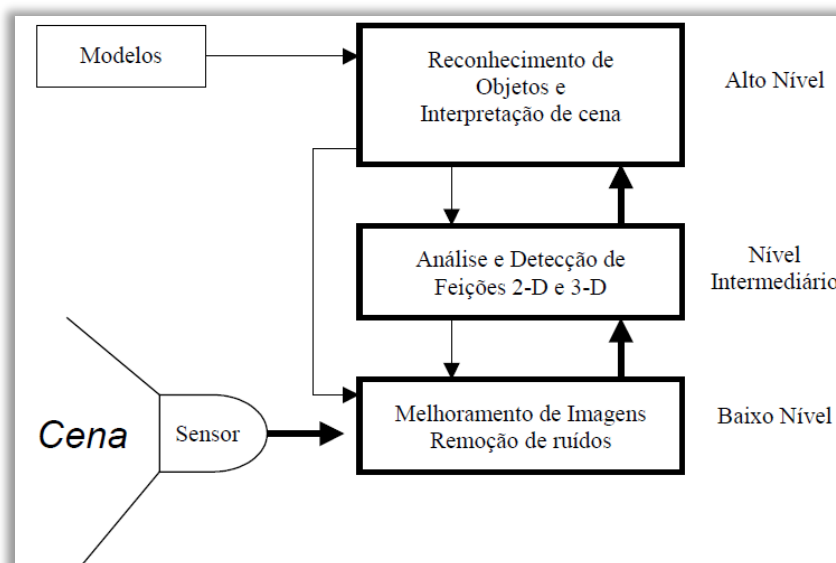


Figura 12 - Hierarquia de tarefas computacionais (MOLZ, 2001 p. 18).

Na figura 12 é possível verificar que as tarefas são divididas em três classes de acordo com o seu propósito (GONZALEZ, WOODS, 2008 p. 26).

- As tarefas do processamento de baixo nível manipulam funções com objetivo de agregar processos para o melhoramento de imagens (pré-processamento), que realizando a remoção de ruídos, alteração de contraste, bem como a extração de características como bordas, formato e tamanho. Aplicada estas tarefas, é possível ter a segmentação da imagem em suas partes com maior relevância, as quais correspondem às linhas ou objetos existentes na cena.
- O processamento pertencente ao nível intermediário consiste na análise das características obtidas no baixo nível a fim de descrever dados simbólicos (bordas e linhas) necessários ao processamento de alto nível. O processamento neste nível tenta estabelecer uma representação adequada de bordas e linhas para que a fusão entre ambas seja significativa.
- O processamento de alto nível busca a interpretação final da cena, com objetivo de reconhecer os objetos e a análise das estruturas relacionais. Também possui o controle das interfaces entre o sistema e seu ambiente.

Os três níveis apresentados acima não são fixos, pois dependem muito da aplicação de visão computacional. Dessa forma, Milano e Honorato (2014, p.2) acreditam que não há um padrão específico para fundamentar a arquitetura de um sistema de visão computacional.

Historicamente, a visão computacional permitiu uma grande evolução tecnológica no desenvolvimento de soluções e pesquisas que buscam um melhor mapeamento de objetos reais,

com o objetivo de proporcionar a representação visual destes objetos. Dentro deste cenário, destaca-se o Microsoft Kinect Sensor.

Nesse sentido, os estudiosos da visão computacional perceberam que a técnica desenvolvida para o reconhecimento de profundidade tridimensional e a câmera *RGB* de espectro visual para reconhecimento visual do Kinect poderiam ser usados para fins da visão computacional (MICHAEL; PAVEL; ANDERS, 2012, p. 1).

Sob o mesmo ponto de vista, Jungong et al (2013, p. 1318) entendem que, por meio do sensor Kinect, é possível melhorar e até mesmo resolver problemas relacionados à visão computacional, os quais podem ser observados na figura 13.

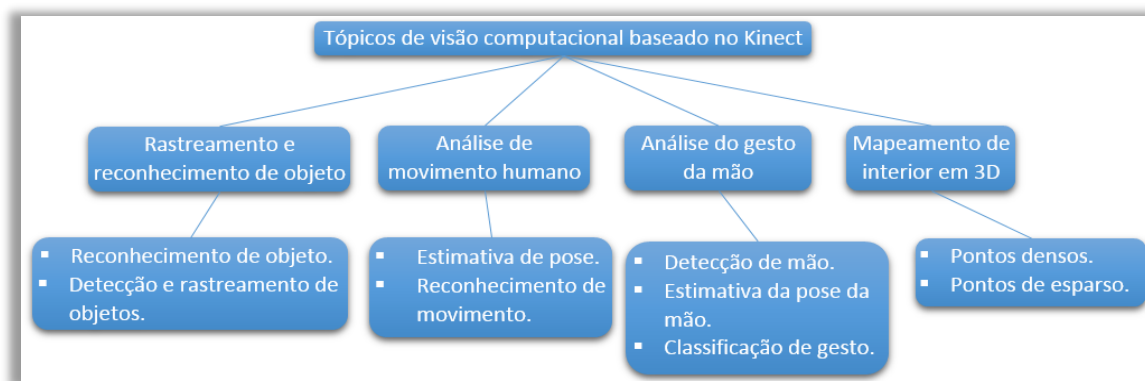


Figura 13 - Tópicos da visão computacional (JUNGONG et al. 2013, p. 1319).

Como pode ser observado na figura 13, o sensor Kinect possibilita a resolução de problemas relacionadas a visão computacional, sendo elas:

- Rastreamento e reconhecimento de objeto: esta etapa apresenta dois processos, sendo o primeiro, a detecção e rastreamento de objetos, e o segundo, o reconhecimento de objeto e cena. O rastreamento de objetos em imagem no padrão *RGB*, a partir de uma cena com fundo estático. O fundo é considerado estático por um certo período de tempo, até que uma imagem de referência seja extraída, servindo de base. No entanto, a detecção de objetos em imagens através de algoritmos não é uma tarefa tão simples. Isto porque existe a possibilidade de mudanças de condições da iluminação e movimentos sutis do fundo interferirem no ambiente. Já o reconhecimento de objeto e cena, ao contrário da detecção de objetos, não fornece a localização do objeto, mas prevê se determinada imagem contém ou não um objeto. O reconhecimento de cena é uma extensão de reconhecimento de objetos, com o objetivo de realizar o mapeamento de toda uma cena. Algoritmos sobre este tema são

baseados principalmente em imagens *RGB*, onde a cor, textura, movimento, ou a combinação de ambas são usados para representar o objeto.

Para a resolução dos problemas o sensor Kinect dispõe do sensor de profundidade e uma câmara *RGB*. O sensor de profundidade utiliza métodos para detectar a profundidade na cena e estabelecer um modelo de fundo mais estável, que é resistente a alterações de iluminação ou à falta de contraste. Já a câmara *RGB* permite estabelecer grande escala de dados em *RGB* devido a sua resolução. Assim combinada às características de *RGB* e de profundidade é possível o reconhecimento de forma eficiente.

- **Análise de movimento humano:** esta etapa divide-se nos processos de estimativa de pose e reconhecimento de movimento. O sensor Kinect dispõe de um inovador método de rastreamento de esqueleto, o que abre novas oportunidades para abordar problemas de análise de movimento humano. Através desse método é possível mapear o corpo humano em posições específicas, podendo assim por meio do reconhecimento de partes do corpo humano criar uma estimativa de pose. Desse modo, pode ser aplicado o método rastreamento de esqueleto no reconhecimento de movimento, ao contrário da estimativa de pose é fato das articulações do esqueleto são obtidas com menor precisão.
- **Análise do gesto da mão:** composta pelos processos de detecção de mão, estimativa da pose da mão e classificação de gesto. A detecção da mão pode ser considerada como um rastreamento de objeto, sendo capturada pela combinação das características de *RGB* e de profundidade. Já estimativa da pose da mão pode ser obtida através de métodos tridimensional, que são baseados em modelo de estimativa de pose, determinado por vetores de parâmetros, bem como por esqueleto da mão. Este último por meio de algoritmos, executa a classificação por pixel e atribui a cada pixel uma parte da mão, e em seguida estima o centro da mão para formar um esqueleto. O processo que compreende os significados dos gestos com as mãos é a classificação de gesto, o qual visa representar técnicas de estimativa para estabelecer uma identificação simples.
- **Mapeamento de interior em 3D:** apresenta os processos em dois tipos de pontos, os densos e por ponto de esparsos. Nos pontos densos todos os pixels de uma imagem contribuem para o alinhamento de cena. Enquanto por pontos de esparsos, os pixels são extraídos a partir de sucessivos quadros e, de forma geométrica em que se encontra na imagem. Com isso, é possível a criação de uma representação digital de um ambiente interno, permitindo a localização automática desse ambiente ou sua reconstrução.

3 MATERIAIS E MÉTODOS

Nessa seção são apresentados os materiais, bem como a metodologia utilizada para o desenvolvimento do trabalho e da aplicação.

3.1 Local e Período

O presente trabalho foi desenvolvido como requisito parcial para a disciplina de “Trabalho de Conclusão de Curso em Sistemas de Informação II (TCC II)” no decorrer do segundo semestre de 2014. O local utilizado para o desenvolvimento foi, basicamente, o Laboratório Banco de Dados e Engenharia de Software (LBDES) em conjunto com o Laboratório de Tecnologia em Saúde (LTS), ambos do Centro Universitário Luterano de Palmas (CEULP/ULBRA), vinculados ao grupo de Pesquisa Tecnologia, Saúde e Qualidade de Vida.

3.2 Matérias

Inicialmente foi realizado uma etapa de estudos acerca dos conceitos envolvidos no trabalho. Para isso, foi utilizado o procedimento metodológico que baseia-se em uma pesquisa bibliográfica, com embasamento em livros, artigos, dissertações de mestrado, teses de doutorado e trabalhos de conclusão de curso.

Na fase de desenvolvimento da aplicação utilizou-se os softwares Visual Studio 2012 para codificar a aplicação, e SQL SERVER 2012 para armazenamento de dados. Também os recursos de *hardware* como um Sensor Kinect do Xbox e um adaptador do sensor Kinect para computador.

A comunicação entre sensor Kinect com a aplicação foi por intermédio dos seus drivers e da biblioteca de classes disponibilizadas no *SDK* na versão 1.8 sendo instalado no *IDE* Visual Studio, conforme apresentado na figura 14.

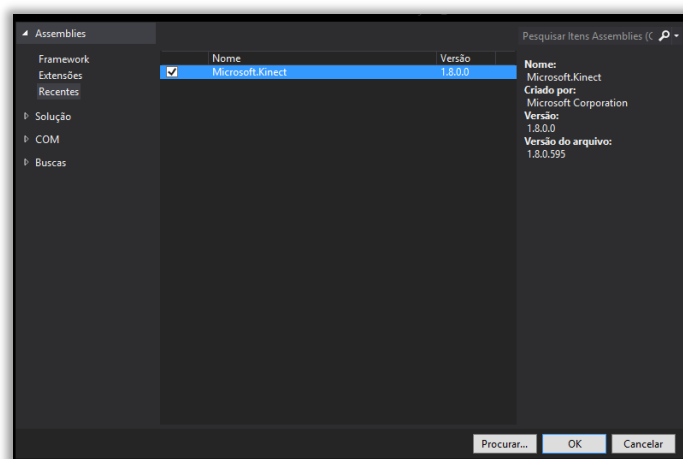


Figura 14 - Referência Microsoft.Kinect.

Como apresentado na figura 14, para adicionar a referência para o *SDK* do Kinect, no projeto selecionar a opção “Add Reference”. Serão apresentadas diversas opções de assembly para adicionar ao projeto, porém, especificamente para este trabalho basta selecionar o “Microsoft.Kinect”.

Os métodos desenvolvidos na aplicação foram implementados na linguagem de programação *C#* (c-sharp) e para a construção das interfaces utilizou-se o padrão *WPF*, que é tipo de projeto proposto no Visual Studio para aplicações Desktop-Windows que utiliza a linguagem de marcação *XAML*.

3.3 Metodologia

A metodologia adotada neste trabalho envolveu inúmeras fases que incluíram atividades relacionadas à pesquisa, à escrita da monografia e ao desenvolvimento da aplicação para a obtenção de atributos antropométricos.

Para Pádua (2004, p. 56) a finalidade da pesquisa bibliográfica é colocar o pesquisador em contato com o que já se produziu a respeito do seu tema de pesquisa. Nesse sentido foram realizadas pesquisas sobre os principais conceitos relacionados ao tema, como a Anatomia Básica Humana, os Pontos Anatômicos, a Antropometria, o Cálculo Estatístico e a Visão Computacional.

As pesquisas foram realizadas com embasamento em livros, artigos, dissertações de mestrado, teses de doutorado e trabalhos de conclusão de curso. Após a realização desta atividade foi possível organizar a estrutura do referencial teórico, bem como realizar a revisão bibliográfica.

Desse modo, a partir do conhecimento adquirido através da pesquisa bibliográfica foi possível prosseguir para o desenvolvimento da aplicação que obtém atributos antropométricos. Para isso iniciou-se o estudo sobre as funcionalidades das classes do *SDK Kinect*, tendo como objetivo compreendê-las para buscar o melhor resultado no desenvolvimento dos métodos da aplicação.

As classes definidas para o prosseguimento do trabalho foram as *ImageStream*, *DepthStream* e *Skeleton*. Através dessas classes e suas subclasses foi possível adquirir a imagem da câmera RGB, o rastreamento da câmera de infravermelho e o mapeamento das coordenadas do esqueleto do paciente.

Antes de iniciar a fase de codificação da aplicação, foi necessário obter os requisitos junto aos profissionais da fisioterapia no Laboratório de Tecnologia em Saúde (LTS) do Centro Universitário Luterano de Palmas – TO (CEULP), como artefatos de cadastro dos pacientes, e quais os atributos antropométricos estáticos a serem capturados como à estatura, largura dos ombros, altura da tíbia, largura dos quadris e comprimento do membro superior.

Posteriormente, foi iniciada a codificação das classes e métodos da aplicação, desenvolvimento das interfaces e a criação do banco de dados.

Após e durante a fase de codificação, foi testada a confiabilidade das medidas obtidas com a utilização da aplicação. Os resultados obtidos foram satisfatórios, com isso é possível concluir que a aplicação desenvolvida é viável de ser utilizada como ferramenta auxiliar aos profissionais de fisioterapia.

Na próxima seção serão apresentados os resultados atingidos com o desenvolvimento deste trabalho.

4 RESULTADOS E DISCUSSÃO

Esta seção trata dos resultados obtidos na realização deste trabalho. Inicialmente, é apresentada a visão geral da aplicação através de sua arquitetura e funcionamento. Em seguida, apresenta-se os métodos dos fluxos de cor, profundidade, esqueleto e, por fim, o detalhamento da forma de obtenção de atributos antropométricos, utilizando o sensor Kinect.

4.1 Visão geral da aplicação

Para possibilitar um melhor entendimento da aplicação, foi elaborada a arquitetura do funcionamento da mesma, conforme a figura 15.

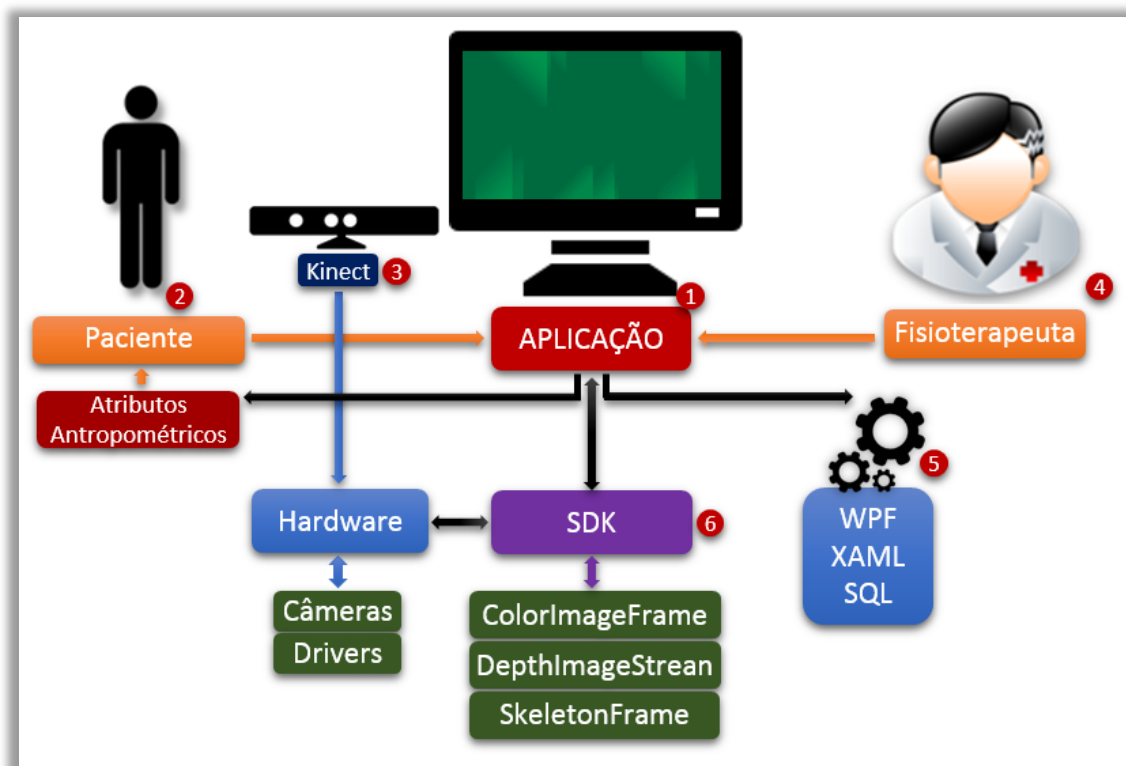


Figura 15 - Arquitetura do funcionamento da aplicação.

Nesta arquitetura, a aplicação (1) é capaz de processar os dados obtidos através do sensor Kinect, sendo necessário que o usuário, no caso o paciente (2), fique posicionado a frente do sensor Kinect (3). Ao realizar a postura, para obter o atributo antropométrico, o profissional fisioterapeuta (4) utiliza as funcionalidades da aplicação para visualizar os dados obtidos. Estes, posteriormente, são processados por meio de métodos providos das classes do *SDK* e por cálculos matemáticas, mais especificamente utilizado o sistema de plano cartesiano.

Para desenvolvimento das interfaces gráficas da aplicação, foram utilizadas algumas ferramentas (5) como: *WPF*, para aplicações Desktop-Windows; *XAML* linguagem de marcação para construção das interfaces; *SQL* para desenvolvimento da base de dados.

Por sua vez, a tecnologia SDK (6) possibilitou a conexão do sensor Kinect com a aplicação por meio de drivers, além de permitir a criação dos métodos de fluxo de cores, profundidade e esqueleto, os quais serão abordados nas próximas seções.

4.2 Fluxo de Cores de Imagem

O sensor Kinect, em sua arquitetura, possui uma câmera que é capaz de capturar imagem colorida de um ambiente, ou seja, obtém a imagem no formato *RGB*. No formato *RGB*, cada *pixel* da imagem possui uma quantidade da cor vermelha, verde e azul, e a soma destes três valores representam a cor do *pixel*. Embora o Kinect trabalhe como uma câmera 3D, em seu nível mais básico, o Kinect é uma câmera padrão que pode capturar transmissões de vídeo usando dados de resoluções e taxas de quadros.

Para trabalhar com o fluxo de cores das imagens é executado um processo com três etapas: primeiramente, o fluxo deve ser ativado, posteriormente a aplicação deve extrair dados de imagem do fluxo e, por último, a aplicação pode processar os dados da imagem. Os últimos dois passos continuam rodando enquanto existir dados de imagem disponíveis (WEBB; ASHLEY, 2012, p. 28).

Como descrito pelos autores, primeiramente o fluxo de imagem colorida deve ser inicializado. O método responsável por esse processo é apresentado na figura 16.

```
51 public void IniciarSensorKinect(KinectSensor kinectSensor)
52 {
53     kinect = kinectSensor;
54     kinect.ColorStream.Enable();
55 }
```

Figura 16 - Método IniciarSensorKinect().

No método `IniciarSensorKinect()`, linha 53, é instanciado o objeto `kinect` da classe `KinectSensor`. Esta tem por finalidade obter acesso aos recursos físicos do sensor Kinect. Em seguida, na linha 54 o objeto `kinect` tem acesso ao fluxo de imagem `ColorStream`, sendo instanciado como o método `Enable` e inicializado para obter o formato de dados da imagem, taxa de quadros e a resolução.

A aplicação obtém os dados de imagem da câmera *RGB* do Kinect através das classes `ColorImageFrame` e `ColorImageStream`, sendo esta estrutura utilizada para aquisição das imagens coloridas (MSDN, 2014, *online*). A classe `ColorImageFrame` fornece acesso aos dados das dimensões, formato e *pixel* para um quadro de cor enquanto que a classe `ColorImageStream` representa o fluxo de cores, contidos nos objetos da classe `ColorImageFrame`. Para melhor entendimento, a figura 16 ilustra o diagrama de classes das classes citadas acima.

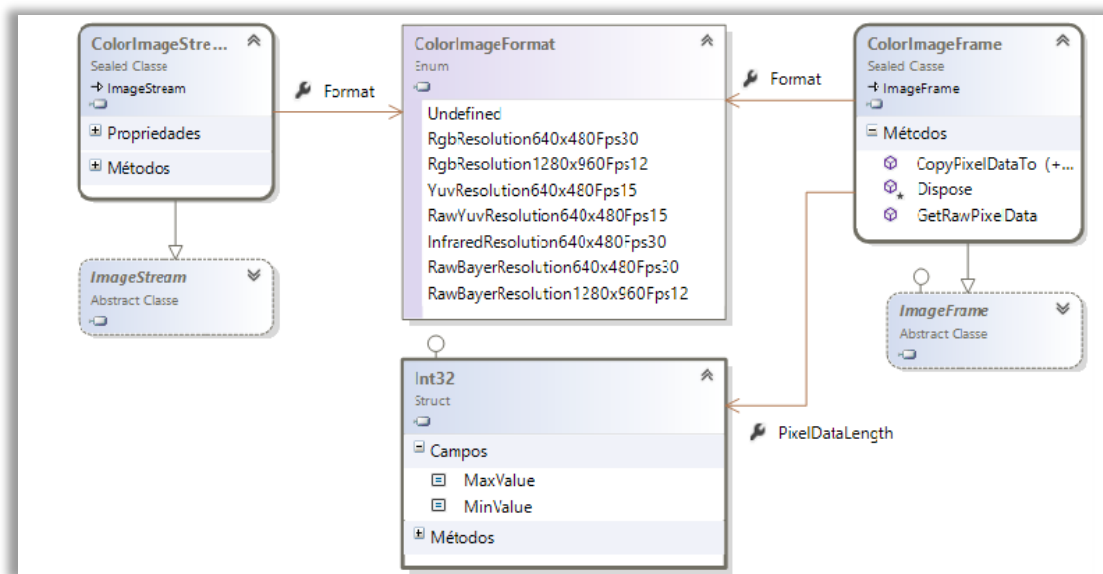


Figura 17 - Diagrama de Classe, classes `ColorImageFrame` e `ColorImageStream` (MSDN, 2014, *online*).

As classes `ColorImageFrame` e `ColorImageStream` são classes derivadas de classes `ImageFrame` e `ImageStream`. Estas classes definem as propriedades e métodos para trabalhar com o quadro de imagem em cores. Abaixo serão descritas as principais relacionadas ao desenvolvimento da aplicação:

- `PixelFormatLength`: é a estrutura responsável por obter o tamanho máximo e mínimo do quadro de dados em *pixel*;
- `ColorImageFormat`: é a propriedade responsável por definir o formato de cor de dados, além de conter os tipos de dados, taxa de quadros e resolução. Conforme apresenta tabela 3.

Tabela 3 - Resoluções e formatos propriedade ColorImageFormat (WEBB; ASHLEY, 2001).

ColorImageFormat	Significado
Undefined	A resolução da imagem não foi determinada
RgbResolution640x480Fps30	A resolução da imagem tem 640x480, os dados do <i>pixel</i> são formato RGB32 a 30 imagens por segundo
RgbResolution1280x960Fps12	Resolução da imagem é 1280x960, os dados do <i>pixel</i> são formato RGB32 tendo 12 imagens por segundo
YuvResolution640x480Fps15	A resolução da imagem é 640x480, os dados do <i>pixel</i> são formato YUV tendo 15 imagens por segundo
RawYuvResolution640x480Fps15	Com resolução de imagem de 640x480, tendo os dados do <i>pixel</i> em YUV em estado natural com 15 quadros por segundo

Com base no entendimento das classes `ColorImageFrame` e `ColorImageStream` para que a aplicação consiga extrair os dados do quadro imagem da câmera *RGB* foi desenvolvido o método `CameraImagemRGB()`. O código que obtém a imagem da câmera *RGB* pode ser visualizado na figura 18.

```

67 private byte[] CameraImagemRGB(ColorImageFrame quadro)
68 {
69     if (quadro == null) return null;
70     using (quadro)
71     {
72         byte[] bytesImagem = new byte[quadro.PixelDataLength];
73         quadro.CopyPixelDataTo(bytesImagem);
74         return bytesImagem;
75     }
76 }

```

Figura 18 - Método `CameraImagemRGB()`.

O método demonstrado acima, objetiva transformar os dados do fluxo de dados oriundos do sensor *RGB* em *bytes* de imagem. Para controle de exceção, na linha 69 é realizada a verificação dos dados do Kinect, para saber se estes são nulos ou não. Na linha 72 é declarado um *array* de *byte* `bytesImagem` que recebe um objeto com tamanho igual a quantidade de *pixels* do *frame* recebido pela propriedade `PixelDataLength`. Em seguida, na linha 73 são copiados os dados de fato do *frame* para o `bytesImagem`. Após os dados serem copiados, na linha 74 o `bytesImagem` contendo os *bytes* da imagem é retornado para o método `Kinect_TodosQuadros()` que é apresentado pela figura 19.

```

81 private void Kinect_TodosQuadros(object sender, AllFramesReadyEventArgs e)
82 {
83     byte[] imagem = CameraImagemRGB(e.OpenColorImageFrame());
84     if (imagem != null)
85     {
86         canvasKinect.Background = new ImageBrush(BitmapSource.Create(kinect.ColorStream.FrameWidth,
87                               kinect.ColorStream.FrameHeight, 96, 96, PixelFormats.Bgr32, null, imagem, kinect.ColorStream.FrameWidth *
88                               kinect.ColorStream.FrameBytesPerPixel));
89     }
90     canvaskinect.Children.Clear();
91     FuncoesEsqueletoUsuario(e.OpenSkeletonFrame());
92 }

```

Figura 19 - Método Kinect_TodosQuadros().

O método ilustrado na figura 19 é responsável por executar uma ação a cada novo frame disponível, ou seja, toda vez que um conjunto de frames estiver apto, ele executa. Para isso, na linha 83 o *array* de *byte* realiza a chamada do método `CameraImagemRGB()`, que retorna os *bytes* que compõem o *frame*. Na linha 84 verifica-se se o *array* de *bytes* que representam o quadro *RGB* está nulo. Para determina que o *background* do canvas inserido na parte visual da aplicação seja preenchido por um *bitmap* e como o *array* de *bytes* vindo do sensor não é um *bitmap* é necessário que seja feita a transformação do *array* em *bitmap* utilizando o código explícito na figura na linha 86.

Logo, a aplicação processa os dados obtidos através dos métodos `CameraImagemRGB()` e `Kinect_TodosQuadros()` e, ao final, apresenta o resultado correspondentes, conforme mostra a figura 20.

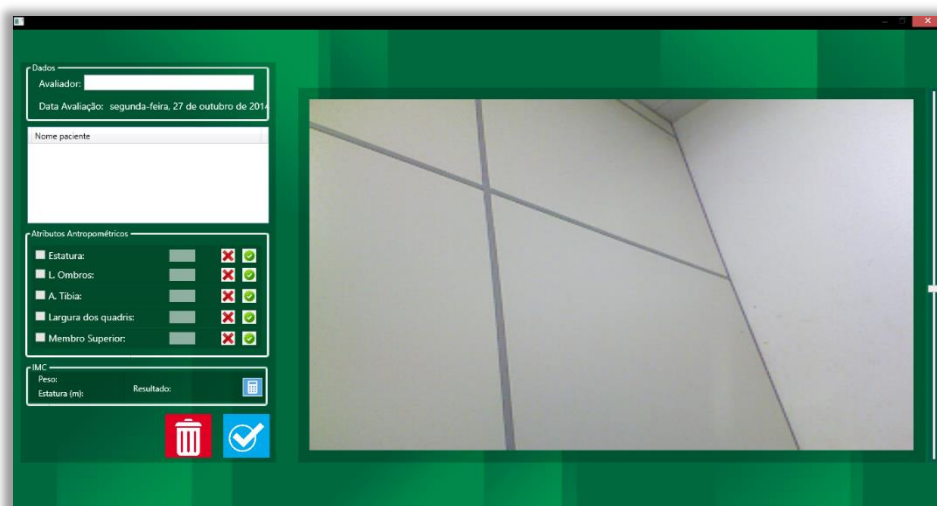


Figura 20 - Interface com Imagem RGB.

A câmera *RGB* permite estabelecer grande escala de dados devido a sua resolução. Porém, o sensor de profundidade utiliza métodos para detectar a profundidade na cena e estabelecer um modelo de fundo mais estável, que é resistente as alterações de iluminação ou à falta de contraste. Assim, combinada às características de *RGB* e de profundidade é possível o reconhecimento de forma eficiente. Na próxima seção do trabalho são apresentadas as características do fluxo de profundidade.

4.3 Fluxo de Profundidade

A seção anterior abordou o funcionamento da câmera *RGB* do sensor Kinect, bem como o método desenvolvido neste trabalho para captura dos dados de imagem em cor. De modo semelhante, é possível capturar os dados do sensor de profundidade do Kinect. No entanto, o princípio de funcionamento do sensor de profundidade bem como as informações devolvidas são inteiramente diferentes ao da câmera *RGB*.

Segundo Leyvand et al., (2011, p. 95), a tecnologia inerente aos atuais sensores de profundidade teve, inicialmente, o objetivo de permitir a interação natural a consoles de vídeo games. Entretanto, as pesquisas que vêm sendo realizadas na área da computação gráfica e da visão computacional têm indicado muitos outros usos interessantes, principalmente no que tange a área da saúde. Por esse motivo, tem sido dada tanta importância ao desenvolvimento de aplicações que promovam o uso desses equipamentos.

Para Jana (2012, p. 121), o fluxo de profundidade é um dos aspectos mais importantes do sensor Kinect. Isto porque representa a sucessão do quadro da imagem de profundidade provenientes do sensor Kinect, pois cada *pixel* de informação dos dados de profundidade representam a distância de um objeto a partir do sensor.

Conforme (MSDN, 2014, *online*), o sensor Kinect possui na sua arquitetura uma câmera profundidade e emissor infravermelho. Estes, trabalham em conjunto para capturar os dados sobre a profundidade dos objetos na frente do sensor. A figura 21 ilustra o campo de visão para captura dos dados de profundidade do Kinect.

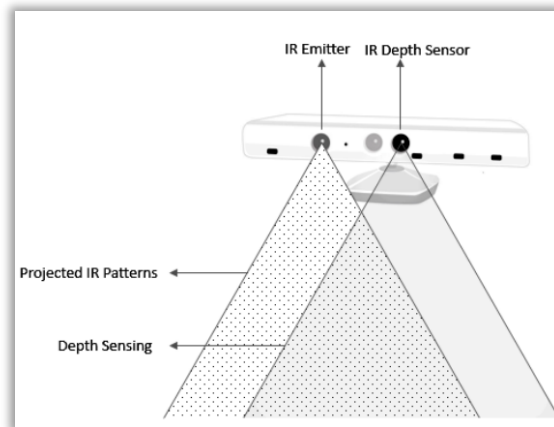


Figura 21 - Campo de visão dos dados de profundidade do Kinect (JANA, 2012, p. 11).

Há dois possíveis modos de escala para os valores de profundidade. No modo padrão os valores variam entre 0,8m e 4m. Já no modo “de perto” (*near mode*), os valores estão entre 0,5m a 3m. Entretanto, este modo afeta diretamente o fluxo de profundidade, pois o Kinect possui uma distância mínima e máxima para que ele possa verificar a distância dos objetos com precisão (CATUCHE, 2012, p. 14).

Para o Kinect, o sensor de profundidade, do mesmo modo do sensor *RGB*, há diversos tipos de formatos de cores de dados, tipos de dados, taxa de quadros e resolução. Existem duas classes principais que são responsáveis por realizar o processamento de dados de profundidade, sendo elas:

- `DepthImageFrame`: é um quadro usado especificamente para imagens de profundidade, fornecendo acesso a dimensões, ao formato e os dados de pixel, e ainda permite o mapeamento de coordenadas entre os quadros de esqueleto e de quadros de cores;
- `DepthImageStream`: representa o fluxo de dados de profundidade da imagem do Kinect, contidos nos objetos da classe `DepthImageFrame`.

De acordo com Webb e Ashley (2012, p.49) a classe `DepthImageStream` realiza o processamento dos dados semelhantemente a classe `ColorImageStream`, apresentada na seção 4.2. Porém, ao invés de criar a quadros com fluxo de cor, criará imagens a partir de um quadro de profundidade. Ambas as classes são derivadas da classe `ImageStream`.

No decorrer dos estudos realizados sobre as classes e métodos do fluxo de profundidade do SDK, foi entendido que a aplicação desenvolvida não necessitaria de métodos que realiza as funções do fluxo de profundidade, pois a classe `DepthStream` é responsável por realizar as funções básicas do sensor IR e conseqüentemente obter os dados de profundidade de relevância

para aplicação. Novamente utilizou-se o método `IniciarSensorKinect()`, conforme ilustra a figura 22.

```
51 public void IniciarSensorKinect(KinectSensor kinectSensor)
52 {
53     kinect = kinectSensor;
54     kinect.ColorStream.Enable();
55     kinect.DepthStream.Enable();
56 }
```

Figura 22 - Método `IniciarSensorKinect()`.

No método `IniciarSensorKinect()`, linha 55, o objeto `kinect` tem acesso ao fluxo de imagem `DepthStream`, sendo instanciado como o método *Enable*. Este, obtém dados como, o formato de dados da imagem, taxa de quadros e a resolução. A figura 23 apresenta a captura do fluxo do esqueleto com apenas a câmera de profundidade habilitada.

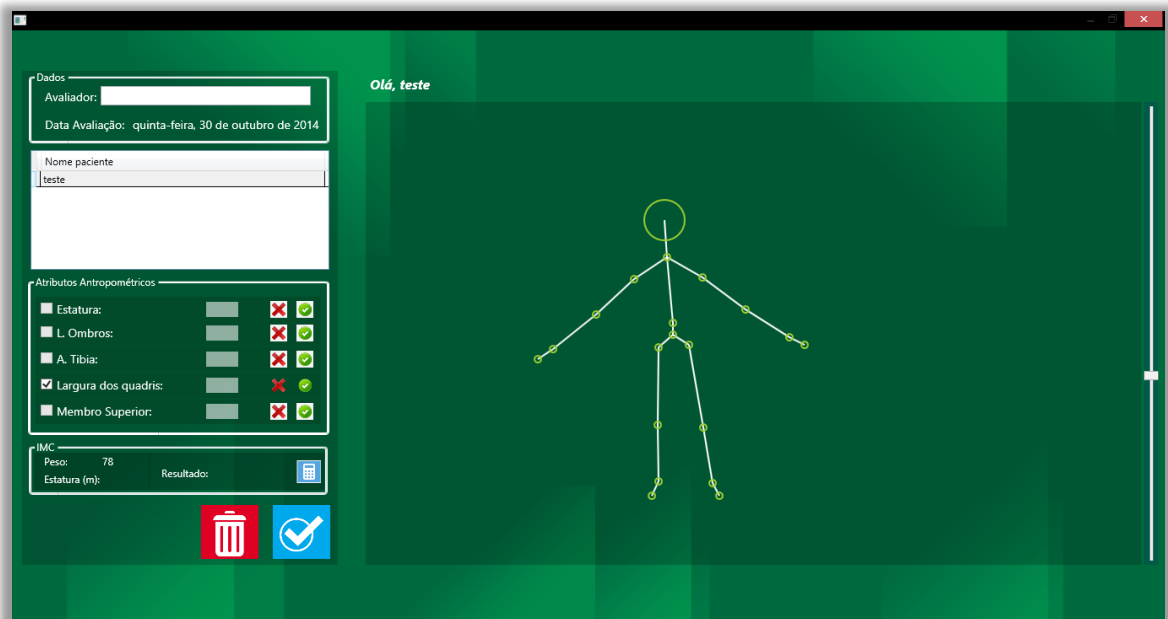


Figura 23 - Câmera de profundidade.

Os dados básicos da profundidade são extremamente importantes para a construção de qualquer aplicação que utilize o sensor Kinect, e neste trabalho não foi diferente. Isto porque, para conseguir realizar o rastreamento do esqueleto, foi necessário usar o fluxo de profundidade. Na próxima seção do trabalho é apresentada as características do fluxo de esqueleto.

4.4 Fluxo de esqueleto

Na seção anterior foi explorado as técnicas do processamento do fluxo de profundidade, bem como o método desenvolvido neste trabalho para capturar dados de imagem em profundidade do sensor Kinect.

Através do fluxo de profundidade é possível ter acesso ao recurso de rastreamento de esqueleto do sensor Kinect. O fluxo de esqueleto é formado por um conjunto de processamentos e de sensores, permitem a identificação de um usuário.

De acordo com Cartuche (2012, p. 21), o termo esqueleto de usuário refere-se a capacidade de percepção do sensor Kinect para detectar um usuário e suas articulações. Isto é possível através do *Motion Capture*, algoritmo baseado em uma rede neural artificial que processa as informações do ambiente em busca de algum usuário (MSDN, 2014, online).

As informações das articulações do usuário são obtidas através dos eixos x , y e z no espaço de coordenadas do sensor Kinect, conforme figura 24.

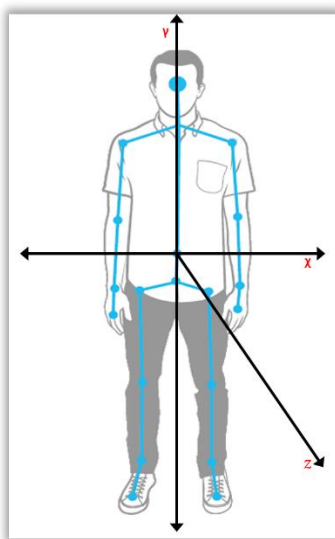


Figura 24 - Eixos do esqueleto adaptado (CATUCHE, 2012, p. 21).

O eixo x estende-se para a direita do ponto de vista do usuário, o eixo y é direcionado para cima e, por fim o eixo z é orientado a partir do sensor para o usuário. Com isso são obtidas as informações de profundidade do usuário.

Com base nesse entendimento, para capturar os atributos antropométricos, inicialmente, tornou-se necessário o estudo sobre a classe `Skeleton`. Essa, juntamente com o sensor de profundidade torna possível realizar o rastreamento esquelético do usuário e,

consequentemente, obter as informações das coordenadas de cada articulação. A figura 25 apresenta o diagrama de classes da classe `Skeleton` e suas respectivas propriedades.

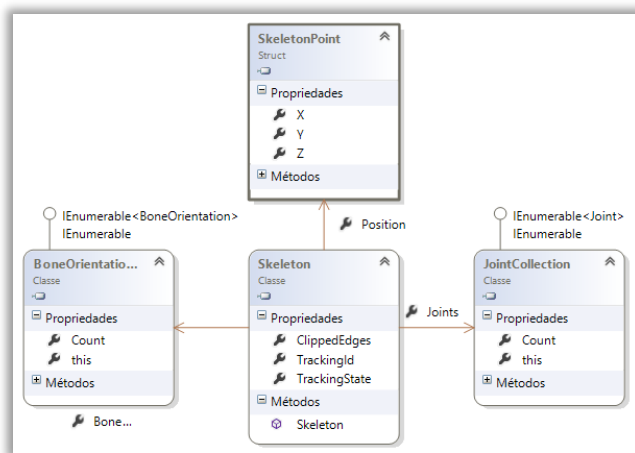


Figura 25 - Diagrama de classe, classe `Skeleton` (MSDN, 2014, online).

Segundo (MSDN, 2014, *online*), as informações do fluxo de esqueleto são obtidas através da classe principal `Skeleton`. Esta é composta por propriedades como:

- `JointCollection`: obtém ou define um conjunto de articulações do esqueleto, referenciados de `Joints`. Cada conjunto contém informações sobre a quantidade de articulações (`count`) no conjunto e o tipo de articulação (`this`);
- `BoneOrientaion`: esta propriedade define a ligação entre as articulações. Também possui as informações sobre a quantidade de ligações e o tipo;
- `SkeletonPoint`: define as posições para cada articulação rastreada. As coordenadas da posição `x`, `y` e `z` para cada articulação são expressas em metros, em vez de milímetros, os quais são utilizados para valores de profundidade.

Diante disso, primeiramente, é necessário obter as informações do rastreamento do esqueleto, para isso, foi desenvolvido o método `FuncoesEsqueletoUsuario()` apresentando na figura 26.

```

32 private void FuncoesEsqueletoUsuario(SkeletonFrame quadro)
33 {
34     if (quadro == null) return;
35     using (quadro)
36     {
37         Skeleton[] skeletons = new Skeleton[quadro.SkeletonArrayLength];
38         quadro.CopySkeletonDataTo(skeletons);
39         IEnumerable<Skeleton> esqueletosRastreados = skeletons.Where(esqueleto => esqueleto.TrackingState ==
SkeletonTrackingState.Tracked);
40         if (esqueletosRastreados.Count() > 0)
41         {
42             Skeleton esqueleto = esqueletosRastreados.First();
43             EsqueletoUsuario funcoesEsqueleto = new EsqueletoUsuario(kinect);

```

Figura 26 - Método FuncoesEsqueletoUsuario().

O método apresentado acima, linha 32, recebe por parâmetro um quadro do tipo `SkeletonFrame`, que contém o quadro de esqueleto adquirido pelo Kinect. No que se refere ao controle de exceção, na linha 34, é realizada a verificação dos dados do Kinect para saber se estes são nulos ou não. Em seguida na linha 37 é declarado uma matriz de `Skeleton` com quantidade de esqueletos presentes no *frame* do Kinect, para isso utilizou-se a propriedade `SkeletonArrayLength`. Após, na linha 38, a matriz `skeletons` recebe todos os dados de cada esqueleto através da propriedade `CopySkeletonDataTo`.

Em seguida é declarado um `IEnumerable<T>`, este, de acordo com a Msdn (2014, *online*), exibe o enumerador que suporta uma iteração simples sobre uma coleção de um tipo especificado. Ao declara-lo na linha 39, conseqüentemente a variável `esqueletosRastreados` receberá todos os esqueletos rastreados da matriz `skeletons`.

Após o rastreamento do esqueleto, tem-se o acesso às informações sobre cada articulação capturada. Como apresentado anteriormente, a propriedade `Joint` dispõe de um indexador com o tipo para cada articulação, conforme ilustra figura 27.

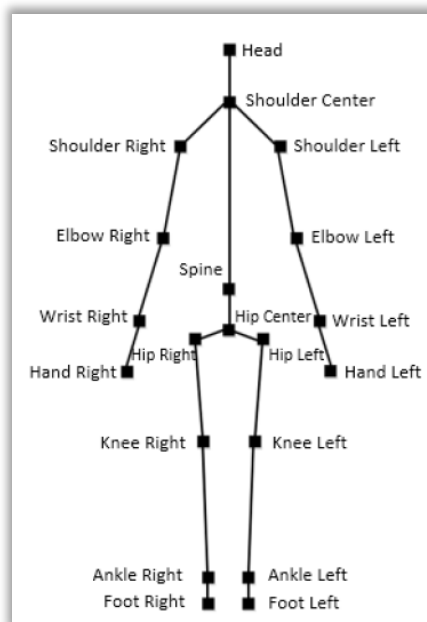


Figura 27 - Conjunto de articulação sensor Kinect (JANA, 2012, p. 163).

Desse modo, ainda pelo método `FuncoesEsqueletoUsuario()`, é possível determinar quais os tipos das articulações e conseqüentemente as posições do eixo x , y , z , conforme apresenta a figura 28.

```

45 Joint primeiraArticulacao = esqueleto.Joints[JointType.Head];
46 Joint segundaArticulacao = esqueleto.Joints[JointType.FootLeft];
47 distancia = DistanciaEntrePontos(primeiraArticulacao.Position.X, primeiraArticulacao.Position.Y, primeiraArticulacao.Position.Z,
segundaArticulacao.Position.X, segundaArticulacao.Position.Y, segundaArticulacao.Position.Z);

```

Figura 28 - Extensão do método `FuncoesEsqueletoUsuario()`.

Na linha 45 e 46 define-se os objetos `primeiraArticulacao` e `segundaArticulacao` do tipo `Joint`, que recebe da matriz `skeletons` o tipo da articulação do esqueleto rastreado. O tipo da articulação é definida pela propriedade `JointType`. Posteriormente, a variável `distancia`, linha 47, definida com tipo `double`, chama o método `DistanciaEntrePontos()`, no qual recebe as posições, de cada articulações, no eixo x y z . Os tipos de articulação utilizado para cada atributo antropométrico foram os:

- Estatura: *Head* e *FootLeft*;
- Largura dos ombros: *ShoulderLeft* e *ShoulderRight*;
- Altura da tíbia: *KneeLeft* e *AnkleLeft*;

- Largura dos quadris: *HipRight* e *HipLeft*;
- Comprimento do membro superior: *ShoulderLeft* e *HandLeft*.

Como visto, o sensor Kinect realiza o mapeamento do esqueleto humano no plano tridimensional. Com isso é possível ter acesso às informações sobre a localização de cada articulação. Tendo essas informações em valores numéricos, segundo Webb e Ashley (2012, p. 291) é possível utilizar diversas expressões matemáticas para o desenvolvimento de soluções. Dentre estas tem-se o cálculo do ângulo entre articulações e distância entre o objeto e o Kinect.

Neste trabalho, para se ter o cálculo da distância tridimensional entre duas articulações, foi utilizado o sistema de plano cartesiano. Este, segundo Bonafini (2012, p. 59) é um sistema de referências formado por duas retas numéricas, sendo uma horizontal e outra vertical que cruzam um determinado ponto. A equação 5 apresenta a fórmula que calcula a distância entre dois pontos do sistema de plano cartesiano.

$$d_{ab} = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2 + (z_b - z_a)^2}$$

Equação 5 - Teorema de Pitágoras.

Com as informações das posições nos eixos x e y entre duas articulações é possível criar uma representação de um triângulo dentro do espaço de coordenadas. Porém, para o Kinect o terceiro ponto, eixo z , pode ser eventualmente utilizado, pois o valor do eixo z não tem efeito significativo sobre o resultado a ser calculado (WEBB; ASHLEY, 2012, p. 291).

Entretanto, para que ocorra uma melhor precisão do resultado, este trabalho optou-se em utilizar as informações do eixo z , pois devido ao deslocamento do paciente em frente ao sensor Kinect pode ocasionar erro na formação do triângulo.

O método `DistanciaEntrePontos()`, foi desenvolvido com o objetivo de calcular a distância entre dois pontos, conforme apresenta a figura 29.

```

54 public static Double DistanciaEntrePontos(Double Ax, Double Ay, Double Az, Double Bx, Double By, Double Bz)
55 {
56     return (Math.Sqrt(Math.Pow((Bx - Ax), 2) + Math.Pow((By - Ay), 2) + Math.Pow((Bz - Az), 2)));
57 }

```

Figura 29 - Método `DistanciaEntrePontos()`.

O método recebe como parâmetros as posições x , y e z , das determinadas articulações definidas anteriormente no método `FuncoesEsqueletoUsuario()`. Com isso é possível aplicar a fórmula do sistema de plano cartesiano tridimensional, conforme apresentado anteriormente na equação 5. Assim sendo, a fórmula corresponde à raiz quadrada da soma do quadrado da diferença das posições x , y e z de A e B.

De acordo com a figura 29, o método `DistanciaEntrePontos()`, na linha 54, utilizou a classe `Math`. Esta, fornece constantes e métodos estáticos para funções trigonométricas, logarítmicas e outras funções matemáticas comuns (MSDN, 2014, *online*). O cálculo é realizado da seguinte forma: A função `Math.Sqrt` retorna a raiz quadrada do resultado obtido pela função `Math.Pow`, esta tem por finalidade retornar o valor elevado à potência especificada, nesse caso elevado ao quadrado.

Por fim, o método retorna o resultado contendo a medida do atributo antropométrico, permitindo assim, ao profissional de fisioterapia, a visualização dos resultados obtidos pela aplicação, como mostra a figura 30.

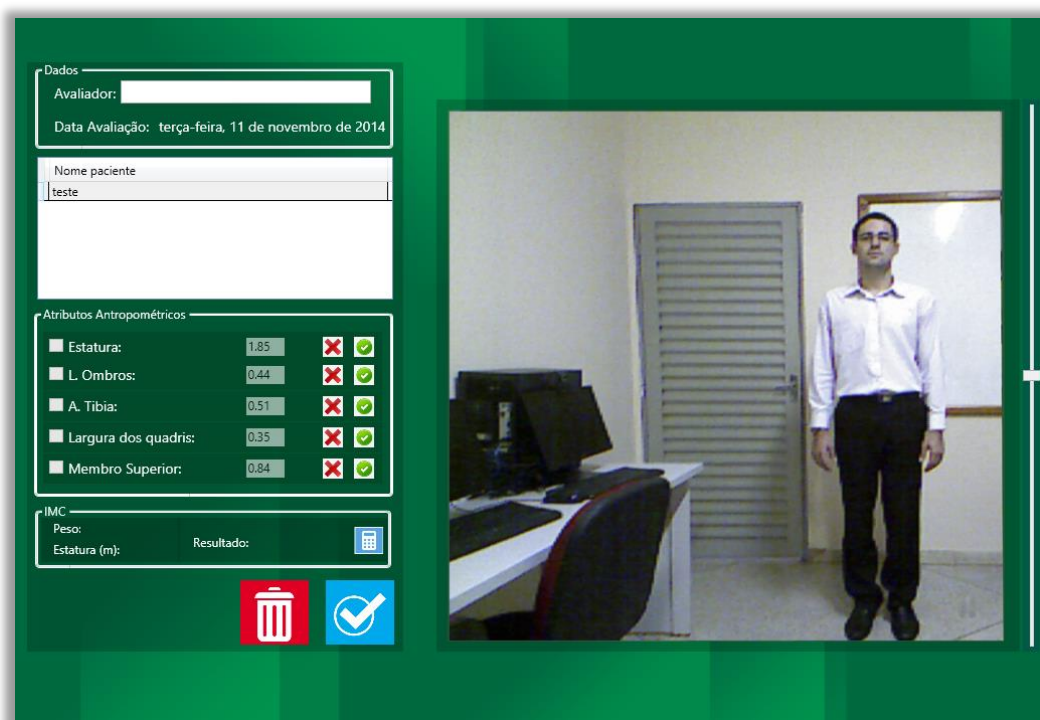


Figura 30 - Interface da aplicação de resultados

A aplicação apresenta ao profissional da fisioterapia os resultados obtidos através dos métodos desenvolvidos neste trabalho. Assim, por meio da aplicação o fisioterapeuta captura,

separadamente, cada atributo antropométrico de um paciente, sendo eles: estatura, largura dos ombros, altura da tíbia, largura dos quadris e comprimento do membro superior.

Diante dos resultados previamente obtidos, ficou constatado que a aplicação através do sensor Kinect, de fato, conseguiu sistematizar o processo de captura dos atributos antropométricos. Entretanto, a precisão dos dados obtidos não inviabiliza o uso da aplicação pois se enquadra na variabilidade das medidas repetidas no mesmo indivíduo.

4.5 Registros de informação

A aplicação disponibiliza ao profissional de fisioterapia a funcionalidade de cadastro de informações do paciente. Isto, é realizado através da interface `cadastro` que dispõem de um formulário que contém campos relacionados aos dados do paciente conforme apresenta a figura 31.

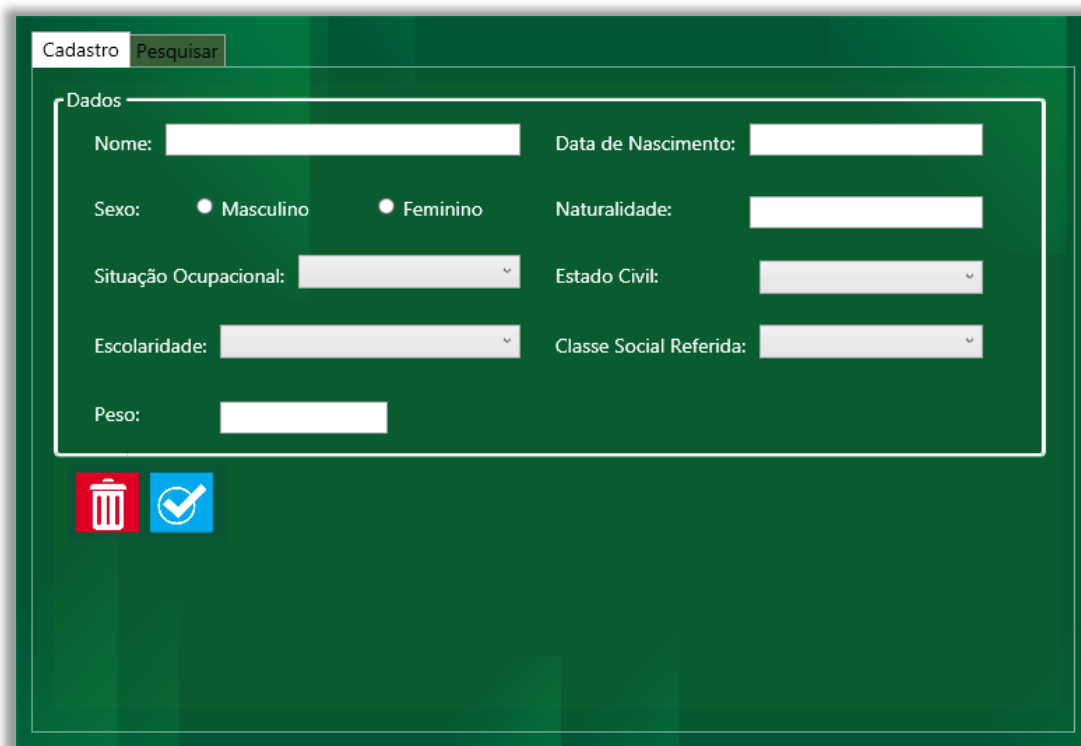


Figura 31 - Interface cadastro paciente.

Na interface ilustrada acima, apresenta o formulário para cadastrar das informações referente ao paciente como: Nome, Data de Nascimento, Sexo, Naturalidade, Situação Ocupacional, Estado Civil, Escolaridade, Classe Social e o Peso. Essas informações são salvas

em um banco de dados utilizado na aplicação, que posteriormente poderá ser visualizado conforme a figura 32.

The image shows a web application interface with a dark green background. At the top, there are two tabs: 'Cadastro' and 'Pesquisar'. Below the tabs, there is a search bar labeled 'Pesquisar:' with a search icon. The interface is divided into two main sections: 'Dados' and 'Dados Antropométricos'. The 'Dados' section contains the following fields: 'Nome:' (text input), 'Data de Nascimento:' (date input), 'Sexo:' (radio buttons for 'Masculino' and 'Feminino'), 'Naturalidade:' (text input), 'Situação Ocupacional:' (dropdown menu), 'Estado Civil:' (dropdown menu), 'Escolaridade:' (dropdown menu), and 'Classe Social Referida:' (dropdown menu). The 'Dados Antropométricos' section contains the following fields: 'Estatura:' (text input), 'Largura dos Ombros:' (text input), 'Altura da Tibia:' (text input), 'Largura dos quadris:' (text input), 'Membro Superior:' (text input), 'Peso:' (text input), 'Avaliador:' (text input), 'Data da Avaliação:' (date input), and 'IMC:' (text input).

Figura 32 - Interface de Pesquisa de Informações.

Com armazenamento das informações do paciente em uma base de dados, posteriormente o profissional da fisioterapia poderá ter acesso às informações sobre os atributos antropométricos realizados em determinado paciente, com isso, poderá realizar avaliações entre as informações obtidas com passar do tempo.

5 CONSIDERAÇÕES FINAIS

Neste trabalho se propôs o desenvolvimento de uma aplicação que use o Sensor Kinect para obter um conjunto de atributos antropométricos. O estudo da antropometria e da anatomia básica humana foram essenciais para o desenvolvimento da aplicação que utiliza o Sensor Kinect para obter atributos antropométricos. Sem este estudo não seria possível desenvolver uma base conceitual sólida que permitisse o início da fase de implementação.

Desse modo, o conhecimento adquirido na etapa de revisão de literatura, bem como os demais assuntos relacionados a área da fisioterapia, tiveram que ser contextualizados para alcançar o objetivo proposto neste trabalho.

Para desenvolver os métodos que realizassem a obtenção de fluxos de imagem colorida, profundidade e o rastreamento do esqueleto foram utilizadas, respectivamente, as classes `ImageStream`, `DepthStream` e `Skeleton` do SDK.

Com a classe `ImageStream` foi possível capturar as informações de imagem colorida de um ambiente, ou seja, obteve a imagem do sensor no formato RGB. Já com a classe `DepthStream`, que obteve acesso as informações do fluxo de profundidade, estas representam a distância de um objeto a partir do sensor Kinect. Por representar a sucessão do quadro da imagem de profundidade provenientes do sensor Kinect, a classe `DepthStream`, é de suma importância para desenvolvimento de qualquer aplicação que utilize o sensor Kinect. Neste trabalho não foi diferente, com base nas informações processadas pela classe foi possível ter acesso ao fluxo de esqueleto do usuário. Este, foi definido pela classe `Skeleton`, que realiza o mapeamento do esqueleto humano no plano tridimensional, dando assim a possibilidade de acesso às informações sobre a localização de cada articulação.

Com isso, foi possível aplicar o sistema de plano cartesiano. Assim, os métodos desenvolvidos neste trabalho realizam um cálculo que obtém a distância entre duas determinadas articulações através do teorema de Pitágoras. O cálculo criado se mostrou funcional conforme esperado. Entretanto, por se tratar do mapeamento do corpo humano em um plano tridimensional pode ocorrer erros relacionados ao fluxo de profundidade, podendo assim, apresentar erros de precisão das medidas antropométricas.

Conclui-se, portanto, que a aplicação para obter um conjunto de atributos antropométricos, por meio do Sensor Microsoft Kinect, possibilita a obtenção da estatura, largura dos ombros, altura da tíbia, largura dos quadris e comprimento do membro superior, sendo estas as características inicialmente definidas para a realização deste trabalho.

Entretanto, este assunto não se esgota, já que trabalhos futuros devem ser realizados para avaliar a precisão dos dados obtidos pela aplicação desenvolvida. Sendo estes relacionados a reconstrução da aplicação utilizando o modelo RGBD, que trabalha com pares de imagens registradas em textura colorida e mapa de profundidade. Isto é, o sensor Kinect realiza o mapeamento do corpo humano em segmentos utilizando a estrutura de dados chamada de árvore de decisão, com isso, é possível criar pontos específicos dentro de cada segmento fornecido pela árvore de decisão.

Assim, acredita-se que seja possível obter mais precisão das medidas e até mesmo aplicar a captura de outros atributos antropométricos. Além disso, outra possibilidade para futuros trabalhos é a utilização de algoritmos de grafos, como as redes bayesianas, para auxiliar na construção de conhecimento probabilístico sobre os dados obtidos pela aplicação.

6 REFERÊNCIAS

- BARROS, Felipe Santos Oliveira; PINHEIRO, Felipe Dos Santos. Cyber guide-dog – help system to the displacement visually impaired people. In: International conference on information systems and technology management - CONTECSI, 10., 2013, São Paulo, 2013. p.3526-3538. Disponível em: <http://infoteca.inf.br/contecsi/smarty/templates/arquivos_template/upload_arquivos/acervo/docs/PDFs/188.pdf>. Acesso em: 19 mai. 2014.
- BASTOS, João Luiz Dornelles; DUQUIA, Rodrigo Pereira. Medidas de dispersão: os valores estão próximos entre si ou variam muito?. **Scientia Medica**, Porto Alegre, v. 17, n. 1. p. 40-44, jan./mar. 2007. Disponível em:<<http://revistaseletronicas.pucrs.br/ojs/index.php/scientiamedica/article/viewFile/1650/1845>>. Acesso em: 09 jun. 2014.
- BRADSKI, Gary; KAEHLER, Adrian. **Learnig Opencv: Computer Vision with the OpenCV Library**. Sebastopol: O'REILLY, 2008. 580p.
- BRENDLER, Clariana Fischer. **Método para levantamento de parâmetros antropométricos utilizando um digitalizador 3d de baixo custo**. 2013. 148 f. Dissertação (Mestrado) – Curso de Design, Universidade federal do Rio Grande do Sul, Porto Alegre, 2013.
- BOUERI, José Jorge. **Antropometria aplicada à Arquitetura, Urbanismo e Desenho Industrial**. São Paulo, FAU/USP, 1991, v. 1.
- DONATI, Marco; CAMOMILLA Valentina; VANNOZZI Giuseppe; CAPPOZZO, Aurelio. Anatomical frame identification and reconstruction for repeatable lower limb joint kinematics estimates. **Journal of Biomechanics**, Roma, v.41, p.2219-2226, 19 maio 2008. Disponível em: <<http://sciencedirect.com/science/article/pii/S0021929008002029>>. Acesso em: 26 mai. 2014.
- GOLDENBERG, Mirian. **A arte de pesquisar: como fazer pesquisa qualitativa em Ciências Sociais**. 4. ed. Rio de Janeiro: Record, 2000. 107 p.
- GONZALEZ, Rafael; WOODS, Richard. **Digital Image Processing**. 3. ed. New Jersey: Pearson Prentice Hall, 2008. 954 p. (978-0131687288).
- HUME, Patria; MARFELL-JONES, Michael. The importance of accurate site location for skinfold measurement. **Journal of Sports Sciences**, Auckland, v. 26, n. 12, p.1333-1340, abr. 2008. Disponível em: <<http://www.tandfonline.com/doi/pdf/10.1080/02640410802165707>>. Acesso: 13 jun. 2014.
- IIDA, I. **Ergonomia: projeto e produção**. 2. ed. São Paulo: Edgard Blucher, 2005.
- JANA, Abhijit. **Kinect for Windows SDK Programming Guide: Build motion-sensing applications with Microsoft's Kinect for Windows SDK quickly and easily**. Birmingham: Packt Publishing, 2012. 366 p. (978-1-84969-238-0).

JÄHNE, Bernd. **Digital Image Processing**. 6. Ed. Berlin: Springer Verlag Berlin Heidelberg, 2005. 606 p. (3-540-24035-7).

JUNGONG, Han; LING, Shao; DONG, Xu; JAMIE, Shotton. Enhanced computer vision with microsoft kinect sensor: A review. **IEEE Transactions on cybernetics**, v.43, n. 5, p. 1318 – 1334, out. 2013. Disponível em < <http://dx.doi.org/10.1109/TCYB.2013.2265378> >. Acessado em: 03 mar. 2013. < http://lshao.staff.shef.ac.uk/pub/KinectReview_TC2013.pdf >.

LEYVAND, Tommer; MEEKHOF, Casey; WEI, Yin Chen; SUN, Jian; GUO, Baining. Kinect identity: Technology and experience. **Computer**, vol.44, p. 94-96, abr. 2011.

LEVIN, Jack; FOX, James Alan; FORDE, David R.. **Estatística para ciências humanas**. 11. ed. São Paulo: Pearson, 2012. 472 p.

LUNET, Nuno; SEVERO, Milton; BARROS, Henrique. Desvio Padrão ou Erro Padrão. Arquivos de Medicina. 2006, Vols. 20, 1-2.

MARQUES FILHO, Ogê; VIEIRA NETO, Hugo. **Processamento Digital de Imagens**. Rio de Janeiro: Brasport, 1999. (8574520098).

MARR, David. **Vision: A Computational Investigation into the Human Representation and Processing of Visual Information**. New York: Freeman, 1982. 429 p.

MICHAEL, Riis Andersen; PAVEL, Lisouski; ANDERS, Krogh Mortensen. **Kinect Depth Sensor Evaluation for Computer Vision Applications: Electrical and Computer Engineering**. Denmark: Department of Engineering, Aarhus University, 2012.

MILANO, Daniel; HONORATO, Luciano Barrozo. **Visão Computacional**. Unicamp, São Paulo, 2010. Disponível em: <http://ft.unicamp.br/liag/wp/monografias/monografias/2010_IA_FT_UNICAMP_visaoComputacional.pdf>. Acesso em: 28 de maio de 2014.

MOLZ, Rolf Fredi. **Uma metodologia para o desenvolvimento de aplicações de visão computacional utilizando um projeto conjunto de Hardware e Software**. 2001. 80 f. Tese (Doutorado) - Curso de Ciências da Computação, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2001.

MORRE, David S.. **Estatística Básica e Sua Prática**. New York: Ltc, 2000. 363 p.

NORTON, Kevin; OLDS, Tim. **Antropométrica: um livro sobre medidas corporais para o esporte e cursos da área da saúde**. Porto Alegre: Artmed, 2005.

PÁDUA, Elisabete Matallo Marchesini de. **Metodologia da pesquisa: abordagem teórico-prática**. 10. ed. São Paulo: Papirus, 2004.

PELEGRINI, Andreia; SILVA, Diego Augusto Santos; PIRES-NETO, Cândido Simões; VIEIRA, Maria Filomena Soares; PETROSKI, Edio Luiz. **O antropometrista na busca de**

dados mais confiáveis. **Revista Brasileira de Cineantropometria e Desempenho Humano**, Florianópolis, v. 13, n. 1, p.82-85, mar.2011. Disponível em: <<http://scielo.br/pdf/rbcdh/v13n1/12.pdf>>. Acesso em: 12 jun. 2014.

PEQUINI, Suzi Mariño. **Ergonomia aplicada ao design de produtos**: Um estudo de caso sobre o design de bicicletas. 2005. Tese (Doutorado em Arquitetura e Urbanismo) - Universidade de São Paulo. São Paulo.

PETROSKI, Edio Luiz. **Antropometria**: técnicas e padronizações. 3. ed. Blumenau: Nova Letra, 2007.

PIRES, David da Silva. **Estimação de movimento a partir de imagens RGBD usando homomorfismo entre grafos**. 2012. 91 f. Tese (Doutorado) - Curso de Computação, Instituto de Matemática e Estatística, São Paulo, 2012.

ROEBUCK, John Arthur; KROEMER, Karl; THOMSON, Walter Gary. **Engineering anthropometry methods**. New York: John Wiley & Sons, 1975.

RON, Larson; BETSY, Farber. **Estatística aplicada**. 4. ed. São Paulo: Pearson, 2004. 638 p.

SILVA, Gustavo Graña Gomes da. **Estudo da utilização do Kinect para biometria**. 2012. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - UNIVERSIDADE FEDERAL DE PELOTAS. Pelotas.

SIRGADO, Tatiana. **Análise cinemática de um movimento humano**. 2010. 23 f. Tese (Doutorado no Curso de Engenharia Biomédica) - Instituto Superior Técnico, Lisboa, 2011.

SILVA, J. C. P.; MARTINS, A. P.; SOARES, J. M. R.; LEITE, M. K.; PASCHOARELLI, L. C. e BOUERI, J. J. Antropometria: uma visão histórica e sua importância para o Design. **Revista Assentamentos Humanos**, Marília, v 9, n 1 p3 - 16, 2007.

TOLEDO, Geraldo Luciano; OVALLE, Ivo Izidoro. **Estatística básica**. São Paulo: Atlas, 1978. São Paulo: 495 p.

WEBB, Jarett; ASHLEY, James; **Beginning Kinect Programming with the Microsoft Kinect SDK**. 1. ed. Apress, 2012.

ZELNIK, Martin; PANERO, Julius. **Dimensionamento humano para espaços interiores**: Um livro de consulta e referência para projetos. Barcelona: Gustavo Gili, 2001. 318 p. (84-252-1835-7).