

Universidade Federal  
do Rio de Janeiro

---

Escola Politécnica

MONITORIZAÇÃO DA FREQUÊNCIA RESPIRATÓRIA COM SOFTWARE  
MODULAR PARA UM SENSOR OPTOELETRÔNICO DE PROFUNDIDADE

Caio Moraes Vahia de Abreu

Projeto de Graduação apresentado ao Curso de Engenharia de Controle e Automação, da Escola Politécnica, Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Bacharel em Engenharia de Controle e Automação.

Orientador: Frederico Caetano Jandre de  
Assis Tavares

Rio de Janeiro  
Agosto de 2015

MONITORIZAÇÃO DA FREQUÊNCIA RESPIRATÓRIA COM SOFTWARE  
MODULAR PARA UM SENSOR OPTOELETRÔNICO DE PROFUNDIDADE

Caio Moraes Vahia de Abreu

PROJETO SUBMETIDO AO CORPO DOCENTE DO CURSO DE  
ENGENHARIA DE CONTROLE E AUTOMAÇÃO DA ESCOLA POLITÉCNICA  
DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE  
DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE  
BACHAREL EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO.

Examinado por:

---

Prof. Frederico Caetano Jandre de Assis Tavares, D.Sc.

---

Prof. Fernando Cesar Lizarralde, D.Sc.

---

Prof. Marcio Nogueira de Souza, D.Sc.

RIO DE JANEIRO, RJ – BRASIL  
AGOSTO DE 2015

Moraes Vahia de Abreu, Caio

Monitorização da Frequência respiratória com software modular para um sensor optoeletrônico de profundidade/Caio Moraes Vahia de Abreu. – Rio de Janeiro: UFRJ/Escola Politécnica, 2015.

XII, 53 p.: il.; 29, 7cm.

Orientador: Frederico Caetano Jandre de Assis Tavares Projeto (graduação) – UFRJ/Escola Politécnica/Curso de Engenharia de Controle e Automação, 2015.

Referências Bibliográficas: p. 52 – 53.

1. Monitorização respiratória. 2. Microsoft Kinect. 3. Software Modular. 4. Período respiratório. I. Caetano Jandre de Assis Tavares, Frederico. II. Universidade Federal do Rio de Janeiro, Escola Politécnica, Curso de Engenharia de Controle e Automação. III. Título.

*Ao meus pais, por me  
incentivarem a curiosidade, pelo  
apoio em minha jornada e  
acreditarem nos meus sonhos.*

# Agradecimentos

Gostaria de agradecer primeiramente aos meus pais, Delcio e Valéria, e ao meu irmão, Rayan, que me deram forças durante toda minha vida. Por acreditarem em meus sonhos e trabalharem duro para me possibilitar de realizá-los. Seja nos momentos ruins e estressantes ou os bons, vocês sempre estiveram presentes me apoiando. Muito obrigado por ter me dado a curiosidade, o interesse, a insistência em sempre fazer o melhor.

Agradeço aos meus amigos e colegas que compartilharam de todos os aspectos do desafio que é a graduação. Em especial a minha turma (T-13): Obrigado por todos os bons momentos e por ter se mantido unida durante tanto tempo!

Agradeço aos professores por todo conhecimento passado e me habilitarem como Engenheiro. Agradeço aos meus ex-orientadores de iniciação: Andreia e Cláudia. Agradeço a CAPES e ao Ministério da Educação pelo programa Ciências sem Fronteiras que possibilitaram uma experiência única em minha vida. Agradeço aos coordenadores do curso de Engenharia de Controle e Automação pelo ótimo trabalho e pelo apoio: Fernando e Afonso.

Agradeço aos colegas que colaboraram com as execuções dos experimentos deste trabalho: Hailton e Raoni. Agradeço ao meu orientador Frederico Jandre por toda a orientação, ajuda, críticas e ideias que tornaram esse projeto possível.

Resumo do Projeto apresentado à Escola Politécnica/UFRJ como parte dos requisitos necessários para a obtenção do grau de Bacharel de Engenharia de Controle e Automação.

## MONITORIZAÇÃO DA FREQUÊNCIA RESPIRATÓRIA COM SOFTWARE MODULAR PARA UM SENSOR OPTOELETRÔNICO DE PROFUNDIDADE

Caio Moraes Vahia de Abreu

Agosto/2015

Orientador: Frederico Caetano Jandre de Assis Tavares

Curso: Engenharia de Controle e Automação

Este trabalho consiste na avaliação do sensor *Microsoft (MS) Kinect®* como instrumento para monitorização de parâmetros respiratórios, especificamente a frequência respiratória (FR). A FR é um parâmetro importante para análise da ventilação pulmonar e pesquisas mostraram que o uso de máscaras e bocais pode afetar o padrão natural da respiração. O uso do *MS Kinect®* como substituto foi testado em trabalhos anteriores quanto à redução do distúrbio causado por máscaras. Com base em um trabalho anterior, foi proposto um novo software com o intuito de automatizar a detecção da região torácica abdominal, compensar movimentos e possibilitar a medição da FR. Como objetivo secundário, o software foi desenvolvido com uma arquitetura modular, possibilitando o desenvolvimento de novas funcionalidades sem alterações drásticas em seu código. Por ser escrito na linguagem C#, o programa não depende de softwares pagos como o *Matlab®* e possibilita uma interface gráfica (GUI) amigável. Experimentos comparando os dados obtidos com o sensor e com um Pneumotacógrafo mostraram ser possível medir a FR em indivíduos de pé e apoiados, para evitar oscilações posturais.

Abstract of Project presented to Escola Politécnica/UFRJ as a partial fulfillment of the requirements for the degree of Bachelor of Control and Automation Engineering.

## MONITORING OF RESPIRATORY FREQUENCY WITH MODULAR SOFTWARE FOR A DEPTH OPTOELECTRONIC SENSOR

Caio Moraes Vahia de Abreu

August/2015

Advisor: Frederico Caetano Jandre de Assis Tavares

Course: Control and Automation Engineering

This study is the evaluation of the sensor *Microsoft (MS) Kinect®* as a tool for monitoring respiratory parameters, specifically the respiratory frequency (RF). The RF is an important parameter for analysis of pulmonary ventilation and research has shown that the use of masks and nozzles can affect the natural pattern of breathing. The use of *MS Kinect®* as a substitute has been tested in previous works on the reduction of disturbance caused by masks. Based on previous work, a new software in order to automate the detection of the abdominal thoracic area, compensate for movements and allow the measurement of RF has been proposed. As a secondary objective, the software was developed with a modular architecture, enabling the development of new features without drastic changes to your code. Written in C# language, the program does not depend on paid software as the *Matlab®* and provides a friendly graphical user interface (GUI). Experiments comparing the data obtained with the sensor and with a Pneumotachograph showed that is possible to measure the RF of individuals standing and supported to avoid postural sway.

# Sumário

<b>Lista de Figuras</b>	<b>x</b>
<b>Lista de Tabelas</b>	<b>xii</b>
<b>1 Introdução</b>	<b>1</b>
<b>2 Fundamentos Teóricos</b>	<b>4</b>
2.1 Cinemática torácica e os ciclos respiratórios . . . . .	4
2.2 Funcionamento do <i>Kinect</i> ® . . . . .	4
2.2.1 <i>ColorStream</i> . . . . .	6
2.2.2 <i>DepthStream</i> . . . . .	6
2.2.3 <i>SkeletonStream</i> e o cálculo do esqueleto . . . . .	7
2.2.4 Sistemas de coordenadas do <i>Kinect</i> ® . . . . .	8
<b>3 Implementação</b>	<b>9</b>
3.1 <i>Algoritmo de detecção de região torácica</i> . . . . .	9
3.2 <i>Algoritmo de cálculo de período respiratório</i> . . . . .	13
3.3 Arquitetura do programa . . . . .	16
3.3.1 A implementação <i>KinMed</i> . . . . .	17
3.3.2 <i>Framework</i> . . . . .	18
3.3.3 Adaptadores . . . . .	19
3.3.4 Algoritmos . . . . .	19
3.3.5 Interface Gráfica (GUI) . . . . .	21
3.3.6 Visão geral da arquitetura e fluxo de dados . . . . .	26
<b>4 Materiais e Métodos</b>	<b>28</b>
4.1 Sensor <i>Kinect</i> ® . . . . .	28
4.2 Pneumotacógrafo . . . . .	28
4.2.1 Calibração do Pneumotacógrafo . . . . .	29
4.3 Metodologia de teste . . . . .	29
<b>5 Resultados</b>	<b>33</b>



<b>6</b>	<b>Discussão</b>	<b>40</b>
6.1	Análise da detecção de ciclos respiratórios e seus períodos . . . . .	40
6.2	Problemas e limitações . . . . .	41
6.2.1	Limitações do <i>Kinect</i> ® . . . . .	41
6.2.2	Limitações dos algoritmos propostos . . . . .	45
6.2.3	Problemas com o <i>SDK/Driver</i> . . . . .	47
6.2.4	Cuidados experimentais . . . . .	47
6.3	Possíveis melhoramentos e trabalhos futuros . . . . .	48
<b>7</b>	<b>Conclusão</b>	<b>50</b>
	<b>Referências Bibliográficas</b>	<b>52</b>

# Lista de Figuras

1.1	Imagem adaptada mostrando a montagem experimental e mapa de profundidade com região de interesse de um dos trabalhos anteriores .	3
2.1	Sensor <i>Microsoft®Kinect®</i> . . . . .	5
2.2	Componentes do sensor <i>Kinect®</i> . . . . .	5
2.3	Posicionamento das juntas do <i>Kinect®</i> sobre o corpo humano. . . . .	7
2.4	Sistemas de coordenadas de imagem e espacial definidos no <i>SDK</i> do <i>Kinect®</i> . . . . .	8
3.1	Três situações onde um objeto no mapa de profundidade pode ter o mesmo valor de profundidade. . . . .	10
3.2	Mapa de profundidade obtido de um sujeito com o <i>Kinect®</i> . . . . .	11
3.3	Plano e sua normal. . . . .	12
3.4	Ilustração do método de rastreamento automático da região torácica abdominal proposto nesse trabalho. . . . .	13
3.5	Funcionamento do algoritmo de frequência respiratória. . . . .	15
3.6	Diagrama lógico do algoritmo de frequência respiratória. . . . .	16
3.7	Visão geral da arquitetura proposta. . . . .	18
3.8	Relação <i>Device</i> , <i>Runtime</i> , <i>RuntimeManager</i> e a aplicação. . . . .	20
3.9	Relação e hierarquia no módulo Algoritmos. . . . .	21
3.10	Tela principal da interface gráfica do <i>KinMed</i> . . . . .	22
3.11	Aba de configuração do sensor <i>Kinect®</i> no <i>KinMed</i> . . . . .	23
3.12	Tela da GUI do algoritmo <i>KinMed.RegionDiscover</i> . . . . .	24
3.13	Tela do algoritmo <i>KinMed.RespirationPeriod</i> . . . . .	25
3.14	Diagrama geral da implementação do <i>KinMed</i> , incluindo alguns módulos. . . . .	27
4.1	Montagem experimental do teste comparativo. . . . .	30
4.2	Diagrama da segunda parte do procedimento experimental. . . . .	31
5.1	Sinais colhidos pelas duas fontes (PT e <i>Kinect®</i> ) antes de tratamento. . . . .	34

5.2	Vazão de ar medida pelo PT versus variação da profundidade média medida pelo ( <i>Kinect</i> ®) e reta de calibração. . . . .	34
5.3	Comparação entre os sinais de vazão de ar e profundidade média medidos pelo PT e pelo <i>Kinect</i> ®, respectivamente. . . . .	35
5.4	Períodos calculados pela rotina em <i>Matlab</i> ®, pelo <i>KinMed</i> e seus respectivos sensores. . . . .	37
5.5	Gráfico de Bland-Altman para os três experimentos. . . . .	38
6.1	Trecho da gravação de vazão de ar e a variação de profundidade média no teste complementar de resolução. . . . .	42
6.2	Período respiratório por tempo calculados pelos três métodos. . . . .	43
6.3	Recortes do gráfico de profundidade média medida pelo <i>Kinect</i> ®. . . . .	44
6.4	Profundidade média do mapa e média dos componentes Z das juntas da região durante o teste de passo. . . . .	45
6.5	Comparativo entre o teste de passo e o sinal calculado pelo <i>Kinmed</i> . . . . .	46

# Lista de Tabelas

2.1	Características visuais do <i>Kinect</i> ® . . . . .	5
5.1	Resultados do teste de respiração espontânea. . . . .	36
5.2	Resultados do teste de respiração controlada, apresentado aqui da forma "Mediana {Min:Max}" . . . . .	39

# Capítulo 1

## Introdução

Um dos parâmetros básicos para a caracterização da ventilação pulmonar é a frequência respiratória (FR), sendo esta utilizada para diagnóstico e avaliação de doenças[1].

A monitorização da FR é usualmente realizada por meios que necessitam de contato com o paciente, como máscaras ou bocais. A influência de sua utilização no padrão respiratório foi evidenciada num estudo realizado por HAN *et al.* [2] na qual se mostrou que a respiração através de máscaras ou bocais altera o padrão respiratório. Em experimentos deste estudo, foi observado que o volume corrente, a frequência respiratória, duração da inspiração e expiração têm seus valores alterados, além do paciente apresentar irregularidades (como pausas ou apneias) com maior frequência.

Uma alternativa para substituir estes instrumentos é o uso de sensores optoeletrônicos para a monitorização sem contato físico com o sujeito. Neste contexto, o sensor *Microsoft*® *Kinect*® ganhou destaque em várias pesquisas e projetos por ter baixo custo, facilidade na aquisição, e utilização em relação a outros sensores mais especializados, além do fato de não utilizar marcadores. Em um estudo conduzido por ALNOWAMI *et al.* [3] foi avaliado que o *Kinect*® tem potencial para aplicações biomédicas por seu baixo custo e desempenho aceitável, com erros de leitura na ordem de milímetros para uma distância entre 85 cm e 115 cm. Trabalhos similares, mas com o objetivo de detecção do movimento torácico e estimativa do padrão respiratório utilizando o *Kinect*® foram realizados por XIA e SIOCHI [4] e AOKI *et al.* [5].

No estudo conduzido por XIA e SIOCHI [4] foi utilizada uma placa sobre o tórax de um voluntário como uma forma de reduzir distúrbios por superfícies irregulares, como dobras de roupas, e ampliar o movimento de respiração a fim de compensar a baixa resolução de profundidade do sensor *Kinect*®. Como resultado, o método proposto obteve coeficientes de correlação na ordem de 0,958 a 0,978, entre o sinal de profundidade do *Kinect*® e o de um sistema de *strain gauge* comercial.

AOKI *et al.* [5] foram além e sugeriram o mesmo tipo de medição sem utilizar uma placa ou contato físico com o sujeito. Neste método, foi utilizada a função de rastreamento de pessoas do *Kinect*® para selecionar a região torácica abdominal, eliminando assim possíveis pontos irregulares como bordas. Estimando as posições dos *pixels* da imagem de profundidade medida pelo *Kinect*® e suas respectivas profundidades, a variação do volume torácico foi calculada através de uma integral numérica de todos os *pixels* medidos. Os dados coletados por este método foram comparados com os medidos por um analisador de gases expirados e apresentaram altos coeficientes de correlação entre os sinais de volume respiratório estimados pelos dois métodos, mostrando novamente a capacidade do *Kinect*® para este tipo de aplicação.

Em sua tese de mestrado no programa de engenharia Biomédica da COPPE/UFRJ, LUSTOSA [6] se baseou nos estudos citados acima para realizar um teste *in vitro* para mostrar que o *Kinect*® pode ser usado para a medição da frequência respiratória. Utilizando-se de um ventilador mecânico, um transdutor diferencial de pressão e um balão de látex, foram feitos experimentos comparando a detecção da frequência respiratória pelo *MS Kinect*® com um método considerado padrão ouro, o pneumotacógrafo (PT) [6]. No experimento, o movimento de inflação e desinflação do balão, gerado pelo ventilador mecânico, é captado pelo *Kinect*® e o volume de ar corrente medido pelo conjunto: transdutor, placa de aquisição e notebook.

Nesse trabalho foi criado um software em *Labview*® (National Instruments, EUA) para captação dos dados de profundidade do *Kinect*®. Estes são salvos em disco e então comparados por um código em *Matlab*® (Mathworks, EUA) com os colhidos pelo padrão ouro. O primeiro quadro de profundidade era inspecionado visualmente na procura da região de interesse. Então, era selecionada a menor possível que incluísse o balão, como mostrado na Figura 1.1. Com a região escolhida, era calculada a derivada da profundidade média de seu interior em cada um dos quadros seguintes, possibilitando assim a medição do tempo entre picos e com isso o período ventilatório.

LUSTOSA [6] chegou à conclusão de que o *Kinect*®, quando comparado ao pneumotacógrafo, se mostra igualmente capaz de detectar as mudanças de fase respiratória e no volume corrente, semelhantemente aos outros estudos citados anteriormente. Porém, em seu trabalho, ele reconheceu que havia limitações em seu código, incluindo o fato de que os movimentos do balão influenciaram a detecção de ciclos e que eles deveriam ser compensados. Por se tratar de um ensaio *in vitro* em vez de uma aplicação prática, pode-se deduzir algumas outras limitações: o ambiente teve que ter condições bem controladas, como por exemplo, uso de uma superfície plana atrás do balão para evitar ruídos, a análise dos dados não foi em tempo real

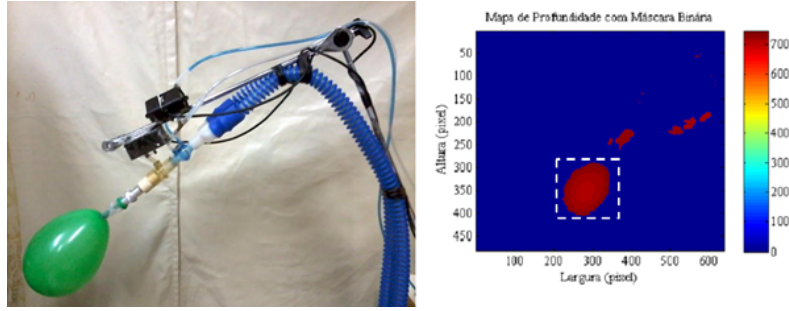


Figura 1.1: Imagem adaptada mostrando a montagem experimental e mapa de profundidade com região de interesse gerada por LUSTOSA [6] em seu estudo.

e a avaliação *in vitro*.

Este trabalho tem como principal objetivo a avaliação do sensor *Kinect*<sup>®</sup> como instrumento para monitorização de parâmetros respiratórios, mais especificamente, o período respiratório e, conseqüentemente, sua frequência. Baseando-se nos estudos anteriores e no algoritmo utilizado por LUSTOSA [6], foi proposto um novo método para o cálculo do período respiratório, procurando melhorar a definição da região de interesse e a compensação de movimentos. Como objetivo secundário, foi desenvolvida uma ferramenta de software para trabalhos na área médica com sensores tipo *Kinect*<sup>®</sup>.

A ferramenta foi testada por meio de um experimento comparativo com o padrão ouro, o pneumotacógrafo, para averiguar seu desempenho. Foram feitos ensaios com indivíduos, utilizando-se tanto uma rotina em *Matlab*<sup>®</sup> quanto o software desenvolvido neste trabalho.

O método descrito acima foi dividido em dois algoritmos que são explicados nas respectivas seções: *Algoritmo de detecção de região torácica abdominal* (3.1) e *Algoritmo de cálculo período respiratório* (3.2). Para atingir o objetivo secundário, uma arquitetura modular é proposta na seção 3.3. Para possibilitar o primeiro objetivo, foram desenvolvidos módulos para esta ferramenta que implementam os dois algoritmos anteriormente mencionados. Os resultados do experimento foram comparados em *Matlab*<sup>®</sup> e comentados na seção 6.1. Por fim, são apresentadas as limitações deste projeto, sugestões para trabalhos futuros e uma conclusão.

# Capítulo 2

## Fundamentos Teóricos

### 2.1 Cinemática torácica e os ciclos respiratórios

O sistema respiratório tem como principal função a troca de gases entre o corpo e o ambiente. É através dos pulmões que o dióxido de carbono, produzido pelas ações metabólicas de nossas células, é removido da corrente sanguínea e ao mesmo tempo o oxigênio, essencial para produção de energia e manutenção das funções celulares, é levado ao sangue.

A respiração pode ser dividida em duas fases: uma **inspiratória** e outra **expiratória**. Na primeira, o pulmão tem seu volume aumentado pela expansão da caixa torácica, diminuindo sua pressão interna em relação ao ambiente e fazendo que os gases atmosféricos sejam transferidos para dentro, onde trocas gasosas acontecem com o sangue. Na segunda, há uma diminuição do volume pulmonar pela retração da caixa torácica, aumentando a pressão e fazendo com que os gases saiam para o ambiente.

Este movimento gasoso é possível graças à expansão e contração dos pulmões: por movimentos de subida e descida do diafragma para aumentar ou diminuir a cavidade torácica, e pela elevação e depressão das costelas para aumentar e diminuir o diâmetro ântero-posterior da cavidade torácica [7]. Com a elevação da caixa torácica, as costelas se projetam quase diretamente para frente, fazendo com que o esterno também se mova anteriormente para longe da coluna, aumentando o diâmetro do tórax em cerca de 20% durante a inspiração.

### 2.2 Funcionamento do *Kinect*®

O *MS Kinect*® é um sensor de rastreamento de movimentos desenvolvido inicialmente para jogos do videogame *Xbox 360*® da mesma empresa. Sua principal função é detectar movimentos dos jogadores posicionados a sua frente, através de mapas de



profundidade, e realizar algum tipo de ação no jogo de acordo com o gesto. Uma foto do sensor pode ser vista na Figura 2.1.



Figura 2.1: Sensor *Microsoft®Kinect®*.

Para realizar tal leitura, o *Kinect®* possui uma câmera RGB, um emissor e um sensor de profundidade infravermelhos (IV), um conjunto de quatro microfones, um acelerômetro e um motor para mudar sua inclinação [8]. Este conjunto de sensores, ilustrados na Figura 2.2, permite a captação de três diferentes tipos de dados: imagem, profundidade e som direcional. A transmissão destes dados é feita através de uma porta USB ligada a um computador com um Kit de Desenvolvimento de Software (*SDK*) instalado. Os sensores do *Kinect®* possuem as características descritas na Tabela 2.1.

Tabela 2.1: Características visuais do *Kinect®*. Adaptado de [8]

Ângulo de visão	43° vertical por 57° horizontal
Ângulo de inclinação vertical	$\pm 27^\circ$
Taxa de quadros (cor e profundidade)	30 quadros por segundo (FPS)

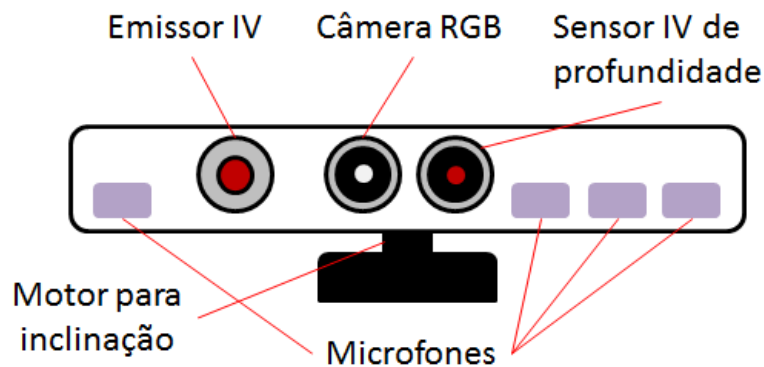


Figura 2.2: Componentes do sensor *Kinect®*.

Neste trabalho foi utilizado o *SDK* oficial implementado na linguagem C# e disponibilizado gratuitamente pela *Microsoft*® em seu site, que constitui-se de ferramentas utilitárias, *drivers* e a biblioteca *NUI* (*Natural User Interface*). Neste trabalho, a biblioteca *NUI* pode ser considerada como uma interface entre o programa principal e o *Kinect*®, disponibilizando os dados coletados por este em classes estruturas e orientadas a eventos. Na *API* da biblioteca *NUI*, existem três classes responsáveis por fornecer os dados de cor (imagem), profundidade e som, além de uma para o cálculo de esqueletos: *ColorStream*, *DepthStream*, *SkeletonStream* e *AudioStream* que serão (com exceção da *AudioStream* que não será utilizada neste projeto) explicadas em seguida.

### 2.2.1 *ColorStream*

Consiste no fluxo contínuo de quadros de cor a uma taxa nominal de 30 quadros por segundo (30 FPS). Cada quadro contém *pixels* de uma imagem capturada pela câmera RGB com resolução máxima de 1280x960 pixels e formato de imagem decidido pelo programador. Este tipo de dado é utilizado neste projeto apenas para visualização.

### 2.2.2 *DepthStream*

Consiste no fluxo contínuo de quadros (mapas) de profundidade igualmente a 30 FPS, podendo assumir resoluções de 80x60, 320x240 e 640x480 *pixels*. Quadros de profundidade carregam uma imagem análoga à de cor; porém, em vez de cada pixel representar dados RGB, eles contém 16 bits, sendo que 3 bits identificam o índice de um jogador (pessoa) e 13 bits para o valor de profundidade da projeção no ambiente daquele pixel até uma superfície de referência.

O método de medição da profundidade de cada ponto é simples: O emissor *IV* emite um padrão pontilhado de raios infravermelhos sobre o ambiente que é refletido por objetos e lido pelo sensor de profundidade *IV*. Além de calcular o valor absoluto de profundidade em milímetros, ele utiliza os ângulos máximos verticais e horizontais do campo de visão para obter a distância vertical e horizontal do ponto à origem do sistema de coordenadas (explicado mais adiante), resultando numa tripla de coordenadas (x,y,z).

O sensor possui uma distância ótima para a leitura de profundidade, que vai de 0,8 metros até 4 metros à sua frente. Fora desta faixa, valores lidos perdem precisão e exatidão. Num estudo realizado por ANDERSEN *et al.* [9], os valores medidos num mesmo plano de profundidade variaram desde aproximadamente 0 até 250 mm para distâncias de cerca de 9 metros. Mesmo mantendo um ambiente estático, existem variações nos valores de profundidade captados da ordem de 1 a

7 mm, com o aumento da profundidade. Observa-se que para o aparelho assumir seu valor final (se estabilizar) são necessários aproximadamente 30 segundos decorridos desde sua ativação [9].

### 2.2.3 *SkeletonStream* e o cálculo do esqueleto

Diferente dos outros *streams*, este utiliza dados de profundidade processados anteriormente para construir um quadro contendo até seis esqueletos de seis pessoas diferentes ao mesmo tempo, sendo dois rastreados mais detalhadamente [8]. Um **esqueleto** é definido neste contexto por um conjunto de 20 juntas e 19 ossos as ligando e uma posição. A posição do esqueleto é determinada pelo centroide das posições das juntas. Cada junta contém a informação de sua posição em relação ao sensor e um nome, sendo este: *HEAD*, *SHOULDER\_CENTER*, *SHOULDER\_X*, *ELBOW\_X*, *WRIST\_X*, *HAND\_X*, *SPINE*, *HIP\_CENTER*, *HIP\_X*, *KNEE\_X*, *ANKLE\_X*, *FOOT\_X*, onde X pode ser *RIGHT* ou *LEFT* para cada lado do corpo. A equivalência do posicionamento destas e a anatomia humana são ilustradas pela Figura 2.3.

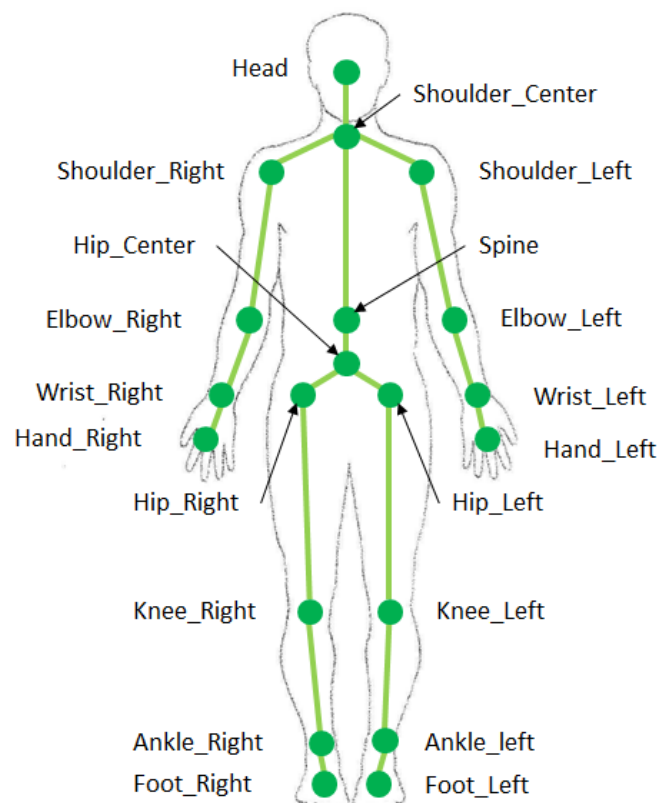


Figura 2.3: Posicionamento das juntas do *Kinect*<sup>®</sup> sobre o corpo humano.

O algoritmo utilizado para determinação do esqueleto é baseado na previsão das posições 3D do corpo numa única imagem de profundidade sem usar nenhuma



Figura 2.4: Sistemas de coordenadas de imagem e espacial definidos no *SDK* do *Kinect*®.

informação temporal, utilizando um método de aprendizagem. Foi utilizada uma abordagem de reconhecimento de objetos: representar as partes do corpo como objetos torna o problema de definição de pose em um problema de classificação de *pixels*, isso gera um algoritmo que consegue estimar partes do corpo invariante a pose, formato ou roupas [10]. O processo constitui-se de identificar as partes do corpo a partir do mapa de profundidade comparando-o com um banco de dados de formas humanas já conhecidas e então estimar a posição das juntas baseando-se no centroide destas.

#### 2.2.4 Sistemas de coordenadas do *Kinect*®

Os sistemas de coordenadas utilizados para localizar um ponto, seja numa imagem ou num espaço 3D, são definidos pelo *SDK* do *Kinect*® e possuem dois tipos distintos: de imagem e espacial.

O sistema de coordenadas de imagem é utilizado para posicionar *pixels* numa imagem. Por se tratar de um mundo bidimensional, os eixos X e Y são perpendiculares e tem origem no canto superior esquerdo da imagem, como pode ser visto na Figura 2.4.a.

O sistema de coordenadas espaciais é utilizado para posicionar juntas e esqueletos no espaço real a frente do sensor. Esse sistema tem três dimensões com eixos cartesianos e origem na posição do *Kinect*®. Conforme se pode observar na Figura 2.4.b, a direção do eixo Z resulta em profundidades maiores quanto mais distante um objeto está do sensor.

# Capítulo 3

## Implementação

### 3.1 *Algoritmo de detecção de região torácica*

Em todos os estudos citados na Introdução foi utilizado um método de extração de uma região de interesse do mapa de profundidade: XIA e SIOCHI [4] utilizaram uma placa para marcar os *pixels* que seriam extraídos do mapa manualmente. LUSTOSA [6] analisou manualmente o primeiro mapa de profundidade e definiu uma região ao redor do balão; AOKI *et al.* [5] utilizaram juntas do esqueleto definido pelo *SDK* do *Kinect®* para definir uma região. Porém, nos métodos escolhidos por XIA e SIOCHI [4] e LUSTOSA [6] não há como realizar leituras em tempo real, pois dependem da análise visual de um pesquisador para definição da região, além do fato do sujeito ter que se manter imóvel durante as leituras. No método realizado no estudo de AOKI *et al.* [5] há uma definição e rastreamento de região automáticos, porém não são compensados eventuais movimentos feitos pelo sujeito, mesmo que involuntários, resultando em uma variação da profundidade média naquela região, podendo ocultar características respiratórias e comprometer a acurácia dos dados.

Considerando as limitações acima, foi proposto neste projeto um novo algoritmo de detecção da região torácica abdominal, com o intuito de defini-la automaticamente e compensar os movimentos do sujeito. Como o mapa de profundidade carrega apenas dados de profundidade do sensor até a superfície de objeto no ambiente, movimentos e deformações de objetos contribuem simultaneamente para sua variação da profundidade. Este fenômeno pode ser caracterizado pela equação 3.1, onde  $\Delta P$  é a variação total da profundidade de um pixel no mapa,  $\Delta P_{\text{movimento}}$  é a variação da profundidade de um pixel devido ao movimento e  $\Delta P_{\text{deformação}}$  é a variação da profundidade de um pixel devido à deformação:

$$\Delta P = \Delta P_{\text{movimento}} + \Delta P_{\text{deformação}} \quad (3.1)$$

Esta característica pode ser visualizada com uma ilustração, na Figura 3.1 po-

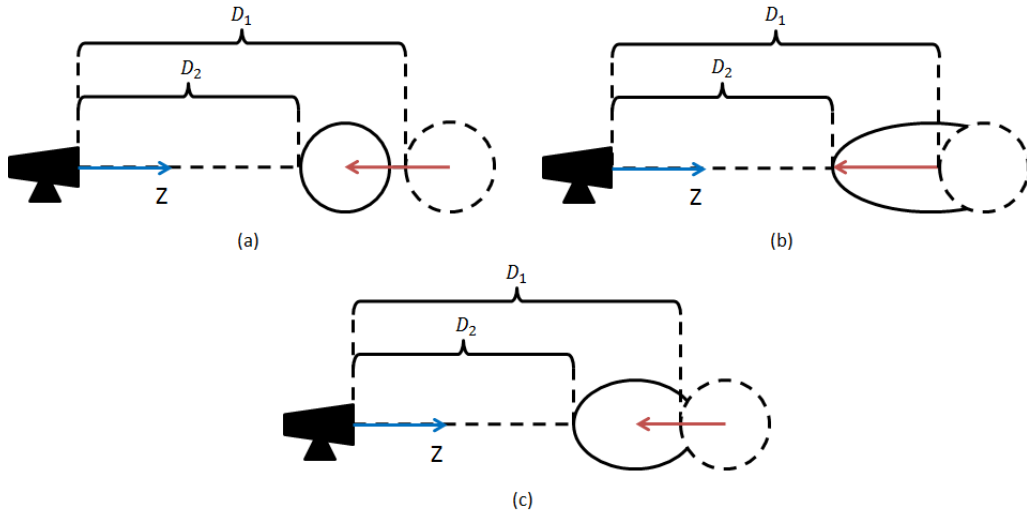


Figura 3.1: Três situações onde um objeto no mapa de profundidade pode ter o mesmo valor de profundidade: a) Movimento do objeto. b) Deformação da superfície do objeto. c) Movimentação e deformação do objeto.

dem ser vistas três diferentes situações nas quais a profundidade de um pixel no mapa pode ter valores iguais: (a) Há um movimento do objeto e a profundidade da superfície muda de  $D_1$  para  $D_2$ ; (b) O corpo se deforma, mudando a profundidade da superfície de  $D_1$  para  $D_2$ ; (c) Há um movimento e deformação do objeto, a profundidade da superfície de  $D_1$  para  $D_2$ . O cenário (c) pode ser considerado uma analogia ao corpo humano durante o ciclo respiratório, onde há uma deformação da caixa torácica, devido à respiração, e ao mesmo tempo o sujeito pode se movimentar.

O principal conceito por trás deste método é eliminar  $\Delta P_{movimento}$  e obter  $\Delta P_{deformação}$  da equação 3.1. Para isso pode-se utilizar dados do esqueleto calculados pelo *SDK* para separar mudanças de profundidade causadas pela respiração e por eventuais movimentos, além de possibilitar a leitura em tempo real. Devem ser escolhidos pontos do corpo que não estejam sujeitos à deformação e ao mesmo tempo formem uma região que contenha a caixa torácica. A decisão dos pontos foi baseada na escolha das juntas no estudo de AOKI *et al.* [5] por conter o tórax e o abdômen. Além disso, observando a Figura 2.3 pode-se notar que as juntas “SHOULDER\_RIGHT”, “SHOULDER\_CENTER”, “SHOULDER\_LEFT”, “HIP\_RIGHT” e “HIP\_LEFT” se localizam em partes do corpo que sofrem pouca deformação pela respiração. Elas correspondem respectivamente a ombro direito, entre ombros, ombro esquerdo, cintura direita e cintura esquerda. Um mapa de profundidade de um voluntário medido pelo *Kinect*<sup>®</sup> nesta região pode ser visto na Figura 3.2:

Utilizando as posições de três destas juntas, é possível calcular um plano que acompanha movimentos e inclinações do corpo. Foram escolhidas as juntas “SHOULDER\_RIGHT”, “SHOULDER\_LEFT” e a posição média entre “HIP\_RIGHT” e “HIP\_LEFT” que serão chamadas de ponto 1 ( $P_1$ ), ponto 2 ( $P_2$ ) e ponto 3 ( $P_3$ ),

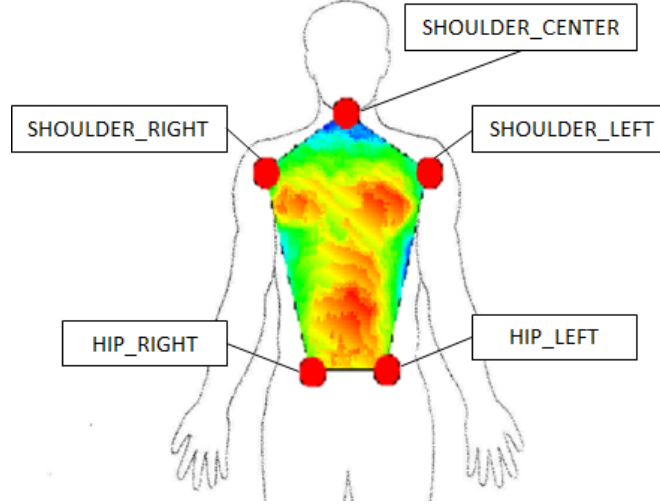


Figura 3.2: Mapa de profundidade btido de um sujeito com o *Kinect*<sup>®</sup>. A Região é formada pelas juntas “SHOULDER\_RIGHT”, “SHOULDER\_CENTER”, “SHOULDER\_LEFT”, “HIP\_RIGHT” e “HIP\_LEFT”.

respectivamente. Em seguida são obtidos três vetores de posição relativa ao sistema de coordenadas do sensor:  $\vec{V}_{s1}$  correspondente a  $P_1$ ,  $\vec{V}_{s2}$  a  $P_2$  e  $\vec{V}_{s3}$  a  $P_3$ , conforme a Figura 3.3.

Subtraindo os vetores  $\vec{V}_{s1}$  e  $\vec{V}_{s3}$  obtêm-se o vetor no plano  $\vec{V}_{31}$  entre os pontos 1 e 3 (Equação 3.2):

$$\vec{V}_{31} = \vec{V}_{s1} - \vec{V}_{s3} \quad (3.2)$$

Similarmente, subtraindo os vetores  $\vec{V}_{s2}$  e  $\vec{V}_{s3}$  resulta no vetor no plano  $\vec{V}_{32}$  entre os pontos 2 e 3 (Equação 3.3):

$$\vec{V}_{32} = \vec{V}_{s2} - \vec{V}_{s3} \quad (3.3)$$

Calculando o produto vetorial entre  $\vec{V}_{31}$  e  $\vec{V}_{32}$ , obtêm-se o vetor normal  $\vec{N}$  do plano passando pelo ponto 3 com coordenadas  $x_0, y_0, z_0$  e os coeficientes  $a, b, c$  e  $d$  da equação do plano [11]:

$$\vec{N} = \vec{V}_{31} \times \vec{V}_{32} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \quad (3.4)$$

$$ax + by + cz = d \quad (3.5)$$

onde:

$$d = -ax_0 - by_0 - cz_0 \quad (3.6)$$

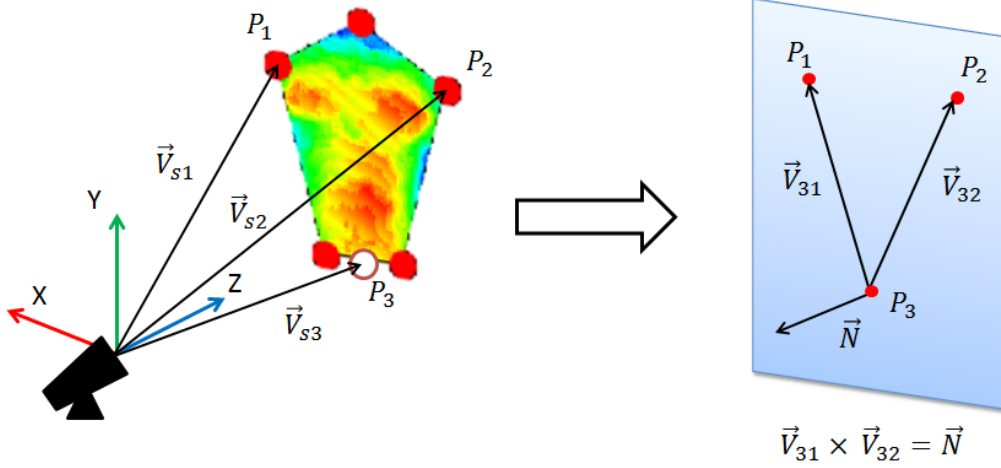


Figura 3.3: Plano e sua normal gerados pelas juntas “SHOULDER\_RIGHT”, “SHOULDER\_LEFT” e a posição média entre “HIP\_RIGHT” e “HIP\_LEFT”.

Projetando-se o vetor  $\vec{V}_{pn}$ , que liga o ponto 3 ( $P_3$ ) até um ponto  $n$  na superfície do mapa de profundidade, na normal  $\vec{N}$  do plano e calculando seu módulo, obtêm-se a distância absoluta  $D_n$  entre o ponto  $n$  e o plano [12] conforme visto na Figura 3.4, ou seja:

$$D_n = |\text{proj}_{\vec{V}_{pn}, \vec{N}}| = \frac{|ax + by + cz + d|}{\sqrt{a^2 + b^2 + c^2}} \quad (3.7)$$

Como o plano se movimenta juntamente com o corpo do sujeito,  $D_n$  não tem seu valor alterado por movimentos, apenas por deformações na superfície. Calculando  $D_n$  para todos os pontos na região de interesse no mapa de profundidade, é possível criar um novo mapa de profundidade, mas em vez de carregar a informação das distâncias entre o sensor e a superfície, contém a distância entre o plano e a superfície, teoricamente eliminando o fator movimento ( $\Delta P_{\text{movimento}}$ ) da equação 3.1. Este novo mapa será referido daqui em diante como “*Mapa da Região Processada*” (MRP). Note-se que mesmo que o sujeito se incline em qualquer eixo, o plano deve acompanhá-lo, deixando  $D_n$  inalterado.

Com isso, o método proposto deverá rastrear a região torácica e abdominal em tempo real, importante para análise do ciclo respiratório, e ao mesmo tempo compensar eventuais movimentos, desde que o plano acompanhe o sujeito. O método descrito acima pode ser ilustrado pelo diagrama da Figura 3.4.



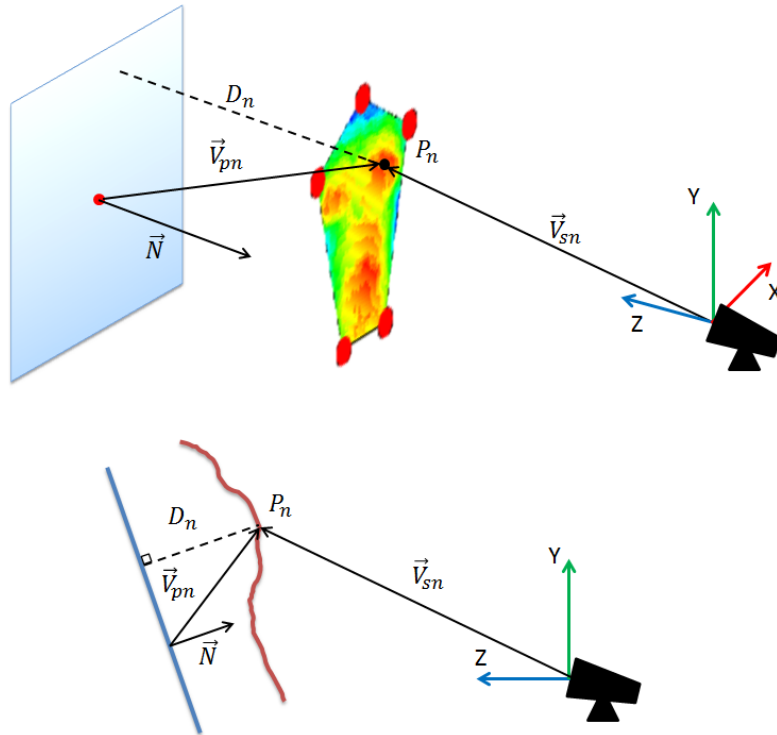


Figura 3.4: Ilustração do método de rastreamento automático da região torácica abdominal proposto nesse trabalho, incorporando compensação de movimentos. Note que a distância  $D_n$  é teoricamente indiferente a translações e rotações do plano.

### 3.2 Algoritmo de cálculo de período respiratório

Este algoritmo tem como finalidade o cálculo do período respiratório, e consequentemente a frequência respiratória, a partir da profundidade média da *MRP* calculado pelo *algoritmo de detecção de região torácica*. Conforme descrito na seção 2.2, o sinal de profundidade do *Kinect*® é essencialmente ruidoso. A amplitude do ruído é da ordem de 2 a 7 mm e a deformação da caixa torácica durante a respiração é da mesma ordem. Consequentemente, isso pode distorcer substancialmente o sinal de profundidade.

Para reduzir o ruído e ao mesmo tempo simplificar os cálculos, foi utilizada a média aritmética de todos os pontos dentro da região do *MRP*, ou seja, a soma de todos os valores de profundidade sobre o número total de pontos dentro da região. Embora se perca informações sobre quais partes do corpo estão se deformando, supõe-se que um sinal da deformação da superfície da caixa torácica seja suficiente para se estimar ciclos respiratórios.

Mesmo após ser calculada a média de profundidade da *MRP*, o sinal resultante ainda possui ruídos e por isso esse sinal foi filtrado. Contudo, apesar do sensor ter uma taxa amostral nominal de 30 quadros por segundo, LUSTOSA [6] mostrou que essa taxa não é constante, sendo necessária uma interpolação para tornar o sinal

adequado para filtragem. Desta forma, este algoritmo é responsável por interpolar o sinal da média de profundidade da *MRP* para uma taxa amostral constante de 30 quadros por segundo (30 Hz) utilizando uma *spline* de 3<sup>o</sup> grau.

Em seguida, o sinal é filtrado por um filtro passa baixa tipo *Butterworth* de 2<sup>a</sup> ordem bidirecional com fase nula. A escolha de um filtro passa baixa se deve ao fato que a frequência respiratória é muito mais baixa, geralmente menor que 1 Hz, do que a do ruído.

A profundidade média, no contexto deste trabalho, pode ser considerada um sinal análogo ao volume respiratório, já que quando a caixa torácica não está expandindo ou contraindo ambos têm valores constantes. Logo, como a vazão de ar pode ser obtida derivando-se o sinal do volume, basta derivar-se o sinal de profundidade média para obter um sinal análogo ao de vazão. Por se tratar de um sinal discreto (amostra a amostra), o cálculo da derivada é realizado utilizando a inclinação da reta secante (método de diferenças finitas) entre o ponto anterior e o atual do sinal de profundidade média. Como um ciclo respiratório tem início na troca das fases de inspiração para expiração ou vice-versa, deve-se procurar cruzamentos por zero no sinal da derivada da profundidade média (*DPM*).

Existem métodos para detecção de cruzamentos de um sinal com zero. Neste trabalho foi escolhido detectar o cruzamento do sinal *DPM* com o resultante do processamento do mesmo com um filtro de média móvel com janela fixa, aqui chamado de *MDPM*. O sinal escolhido constitui-se da subtração de *DPM* por *MDPM* e quando o resultado é zero, aquele instante no tempo é considerado um ponto candidato. Contudo, o interesse está apenas na troca inspiração/expiração ou na expiração/inspiração, sendo a última escolhida para este projeto. Com isso, apenas os pontos candidatos nos quais há passagem de valores positivos para negativos são considerados válidos. Este algoritmo pode ser ilustrado pela Figura 3.5, onde os pontos vermelhos representam cruzamentos válidos. Este método tem a vantagem de tornar a detecção de cruzamentos menos sensível a ruídos de baixa amplitude e alta frequência e ao fato de que pode ser modificado para detecção em tempo real.

Apesar da capacidade deste algoritmo de trabalhar em tempo real (quadro a quadro), foi optado pela sua implementação *off-line*. Para realizar a interpolação por *spline* e a filtragem de ida e volta é necessário um conjunto de dados. Em "tempo real" isso seria possível se fossem analisados dados em bateladas, gerando atrasos que podem comprometer a qualidade da análise *on-line*, sendo necessário mais estudos para validar seu uso. Mesmo com a existência de métodos melhores e mais robustos, o algoritmo foi implementado desta forma para permitir que futuros trabalhos o tornem *on-line*.

Para evitar períodos muito curtos devido a cruzamentos de alta frequência gerados por ruídos ou falsos picos nos valores de *DPM*, o algoritmo conta com algumas

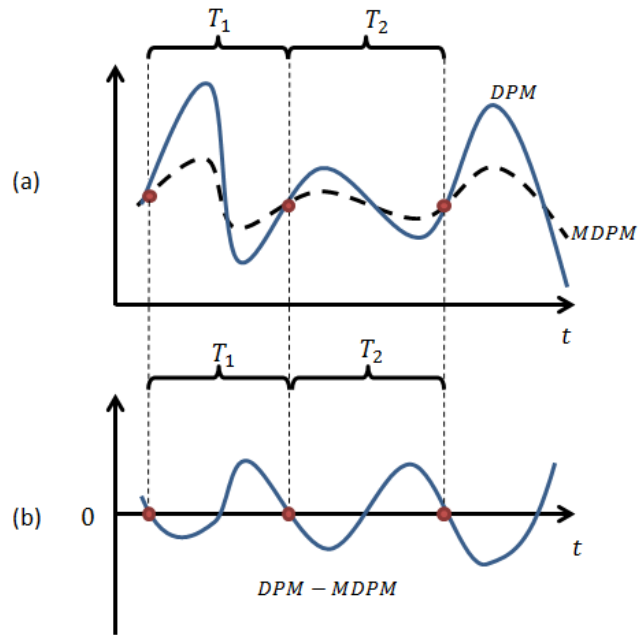


Figura 3.5: Funcionamento do algoritmo de frequência respiratória. O método consiste em identificar cruzamentos entre a derivada da profundidade média ( $DPM$ ) e seu sinal filtrado ( $MDPM$ ), gerando pontos onde há um início de ciclo (pontos vermelhos). (a)  $DPM$  cruzando a  $MDPM$ . (b) Sinal gerado pela subtração da  $DPM$  pela  $MDPM$ .  $T_1$  e  $T_2$  são exemplos de períodos detectados neste algoritmo.

regras adicionais para considerar cada cruzamento como verdadeiro. O diagrama lógico da Figura 3.6 mostra como o algoritmo detecta estes pontos. Um ponto é válido quando: valores anteriores são negativos *ou* um limiar mínimo de diferença é atingindo *ou* o suposto período deve ser maior que um valor mínimo ajustável. Ao final, a diferença entre esses dois pontos no tempo é calculada e o resultado é o período respiratório.

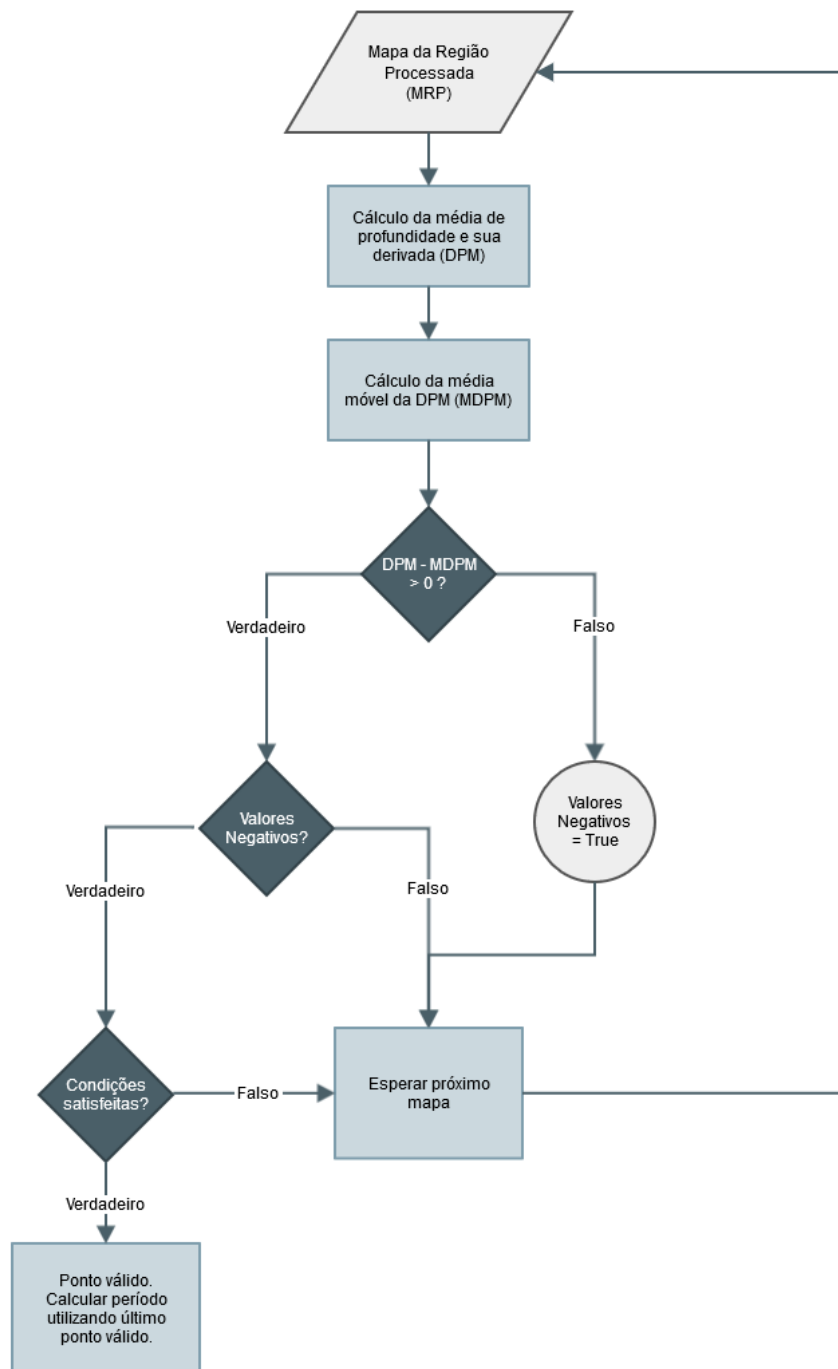


Figura 3.6: Diagrama lógico do algoritmo de frequência respiratória.

### 3.3 Arquitetura do programa

Neste projeto, a implementação dos algoritmos propostos nas seções 3.1 e 3.2, foram planejadas levando em conta os três seguintes objetivos:

1. Criação de uma aplicação prática e com interface amigável, que possibilite ser utilizada em ambientes fora do laboratório em computadores não necessariamente de alto desempenho.
2. Ser expansível, possibilitar a adição de novas funcionalidades sem que haja, necessariamente, alterações significativas no código.
3. Não depender do modelo de sensor, ou seja, ser possível sua troca sem mudanças significativas no código.

Para satisfazer ao primeiro objetivo, a implementação do projeto foi realizada na linguagem de programação C#, uma linguagem moderna e derivada da plataforma .NET da *Microsoft*®. Esta linguagem tem como características a ênfase em orientação a objetos, a rapidez de programação e ao bom desempenho. A escolha se deu também pelo fato do *SDK* oficial do *Kinect*® ser escrito e oferecer suporte completo nesta linguagem.

Quanto à interface gráfica, foi escolhida a biblioteca *WPF* da *Microsoft*®, nativa em C#. Ela oferece recursos modernos e aparência amigável, além de ter como modelo de programação a separação interface gráfica e a lógica programável [13], que possibilita a flexibilidade desejada nos outros objetivos.

Para completar o segundo objetivo, foi utilizado o conceito do design modular em programação. O design é uma técnica que consiste em separar as funcionalidades de um programa em módulos independentes intercambiáveis, onde cada um contém toda a lógica para executar apenas a funcionalidade desejada [14]. Este método também traz benefícios para o desenvolvimento, tornando possível que diferentes desenvolvedores programem módulos independentemente um do outro e sem modificar a aplicação principal.

Para satisfazer o objetivo 3 e possibilitar o desenvolvimento independente, é necessário que haja uma separação da lógica de programação dos elementos do *SDK* do *Kinect*®. Isso foi atingido utilizando o conceito de *framework*, que será detalhadamente explicado na próxima seção.

Com os objetivos definidos e os conceitos de programação já mencionados, foi desenvolvida a arquitetura apresentada na Figura 3.7. Ela consiste em quatro componentes abstratos principais: *Framework*, adaptadores, algoritmos e a interface gráfica. Esses módulos e suas implementações serão explicados detalhadamente em seguida.

### 3.3.1 A implementação *KinMed*

Neste trabalho, foi criada a aplicação **KinMed**, junção das palavras *Kinect* + *Medical*. As seguintes sessões irão detalhar como os conceitos anteriores foram aplicados

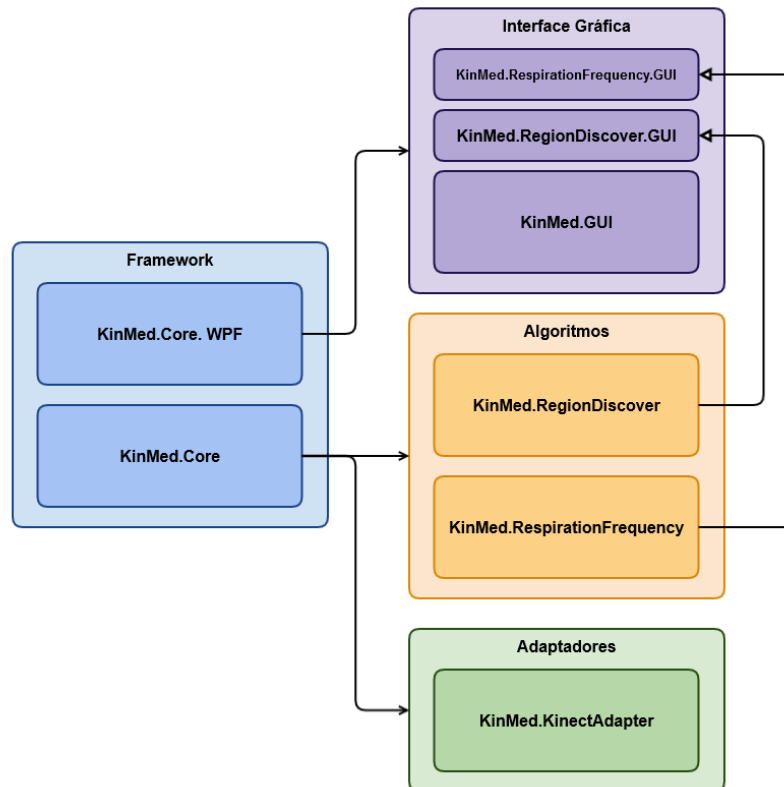


Figura 3.7: Visão geral da arquitetura proposta.

ao programa.

### 3.3.2 *Framework*

Um *framework*, em desenvolvimento de software, é uma abstração que une códigos comuns entre vários projetos de software, provendo uma funcionalidade genérica. No presente contexto, este componente consiste de tipos básicos, interfaces, classes abstratas, utilitários e enumerações. O *framework* define assim um formato de dados padrão em que todos os componentes que obedecem a suas interfaces podem entender e escrever. Desta forma existe uma linguagem padrão e comum, permitindo a criação de módulos independentes, um dos nossos objetivos.

Com o intuito de desenvolver um software que separa a camada lógica e a de interface gráfica de usuário (GUI), o *framework* foi dividido em duas partes:

- **KidMed.Core:** Contém classes abstratas e interfaces de programação. Responsável por definir quadros, esqueleto, dispositivos (sensores) bases, gerenciadores de dados, operações matemáticas, processamento de sinais, etc.
- **KinMed.Core.WPF:** Contém classes e utilitários para a aplicações em *WPF* da *Microsoft*® como: Gerenciadores de janelas e geradores de visualização.

Como a Figura 3.7 mostra, módulos pertencentes aos componentes *Algoritmos* e *Adaptadores* utilizam-se do *KinMed.Core*, por exemplo, para definir suas implementações. Por obedecer às interfaces e utilizar-se das classes padrões, todos os dados de quadros, esqueletos e tipos matemáticos gerados por estes módulos podem ser lidos por outros módulos e vice-versa, bastando que internamente seja programada a lógica de conversão entre um formato específico e o padrão definido pelo *framework*. Esta função será exemplificada com o adaptador *KinMed.KinectAdapter*.

O componente *KinMed.Core.WPF*, além de fornecer ferramentas para a interface gráfica *WPF*, possibilita que outros módulos forneçam suas próprias interfaces gráficas para serem mostradas na janela principal. Por exemplo, abas de configuração para cada sensor e abas de visualização para algoritmos. Isto permite que os desenvolvedores destes componentes possam criar seu próprio modo de visualização de dados, além de tornar sua programação independente da *GUI* principal.

### 3.3.3 Adaptadores

Adaptadores são os componentes que separam o *SDK* do sensor e o restante do programa. Eles são responsáveis por converter os dados fornecidos pela *SDK* para o formato padrão definido pelo *framework*, assim tornando outros módulos independentes de um sensor específico.

As implementações desse módulo são chamadas de *Runtime*, responsáveis pelo gerenciamento, enumeração e ciclo de vida dos respectivos sensores a ele atribuídos. Através do *Runtime*, a aplicação pode identificar um sensor, iniciá-lo ou pará-lo. Um *Runtime* pode ser responsável por vários sensores conectados ao mesmo tempo, porém apenas de um tipo. O programa permite a conexão de mais de um *Runtime* ao mesmo tempo. Para criar um encapsulamento do *driver* de um sensor, existe ainda a classe abstrata chama *Device*, contendo comandos comuns e conversão de dados. Para este projeto, foi desenvolvido um *Runtime* do sensor *Kinect®*, o ***KinMed.KinectAdapter***

A relação *RuntimeManager*, *Runtime*, *Device* e o sensor pode ser ilustrada pelo diagrama da Figura 3.8. Nele pode-se notar o comportamento descrito acima: cada sensor é encapsulado por um *Device* e este pertence ao respectivo *Runtime* que por sua vez é gerenciado pelo ***RuntimeManager***.

### 3.3.4 Algoritmos

Um algoritmo é chamado, neste trabalho, de um conjunto de métodos e classes criados para analisar, processar ou exibir dados disponibilizados pelas implementações de *Device*. Esta estrutura traz a separação entre diferentes lógicas e introduz modularidade e independência no projeto. Assim, a implementação de um algoritmo pode

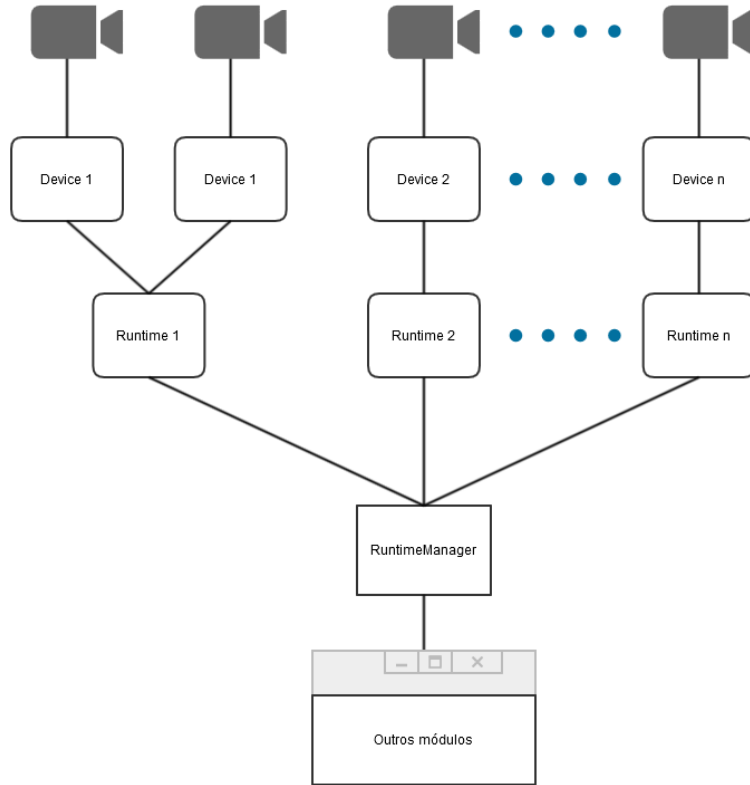


Figura 3.8: Relação *Device*, *Runtime*, *RuntimeManager* e a aplicação.

ser feita separadamente da implementação de outro. Esta metodologia permite, por exemplo, a troca de um algoritmo A por um algoritmo B sem que haja necessidade de modificações significativas em B.

Similar aos adaptadores, há uma hierarquia e um gerenciador para esse módulo, porém, com relações diferentes. Instância do algoritmo é o equivalente a um *Device*. Um algoritmo pode ter diferentes instâncias ativas ao mesmo tempo e receber dados de nenhum ou de múltiplos *Devices*, receber dados de outros algoritmos ou até mesmo gerar dados como um *Device*. Diferentes instâncias têm a mesma lógica e processamento de dados; porém, podem ter distintas configurações e receber dados de outros *Devices* ou algoritmos.

Análogo aos *Runtimes* existe o *Module* de um algoritmo, o qual é responsável pelo ciclo de vida, iniciação/finalização de instancias de algoritmos e acesso a estas. Diferentemente do *Runtime*, ele pode criar instâncias arbitrariamente. A Figura 3.9 ilustra a relação *Instancia*, *Module* e *AlgorithmManager*, o gerenciador de algoritmos.

Para os objetivos desde projeto, dois Módulos de Algoritmos foram desenvolvidos.



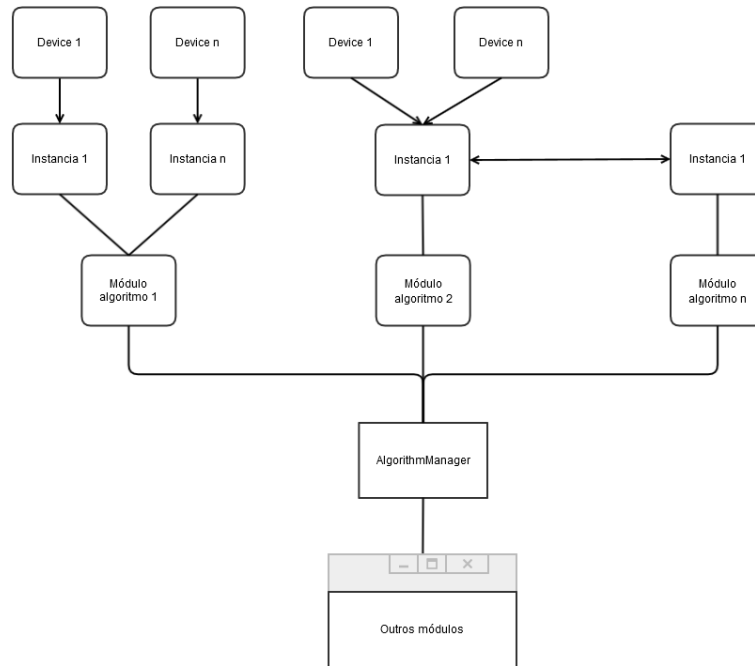


Figura 3.9: Relação e hierarquia no módulo Algoritmos.

### KinMed.RegionDiscover

Analisa o quadro de profundidade juntamente com o de esqueleto para calcular a região torácica e a profundidade média da mesma em tempo real (quadro a quadro), utilizando-se do *Algoritmo de detecção de região torácica* explicado na seção 3.1. Permite que estes dados sejam gravados em disco no formato *Comma Separated Values (.csv)*. Provê para outras instâncias um novo quadro de profundidade contendo as informações processadas. Permite ajustes (*off-set*) dos pontos do esqueleto utilizados para detectar regiões, assim esta pode ser ajustada de pessoa para pessoa.

### KinMed.RespirationPeriod

Processa dados da profundidade média gerados e salvos em disco pelo *KinMed.RegionDiscover*, identificando o período respiratório por meio do o algoritmo proposto na seção 3.2. É altamente configurável quanto aos coeficientes do filtro *Butterworth*, ao de média móvel e a seleção da faixa de dados. Permite exportação de dados dos períodos respiratórios calculados no formato *Comma Separated Values (.csv)*.

## 3.3.5 Interface Gráfica (GUI)

Este módulo constitui a camada mais elevada da aplicação, a interface homem programa, em que os dados disponibilizados pelos *Devices* e pelos algoritmos podem ser visualizados e interagir com o usuário. A principal função desde módulo é a

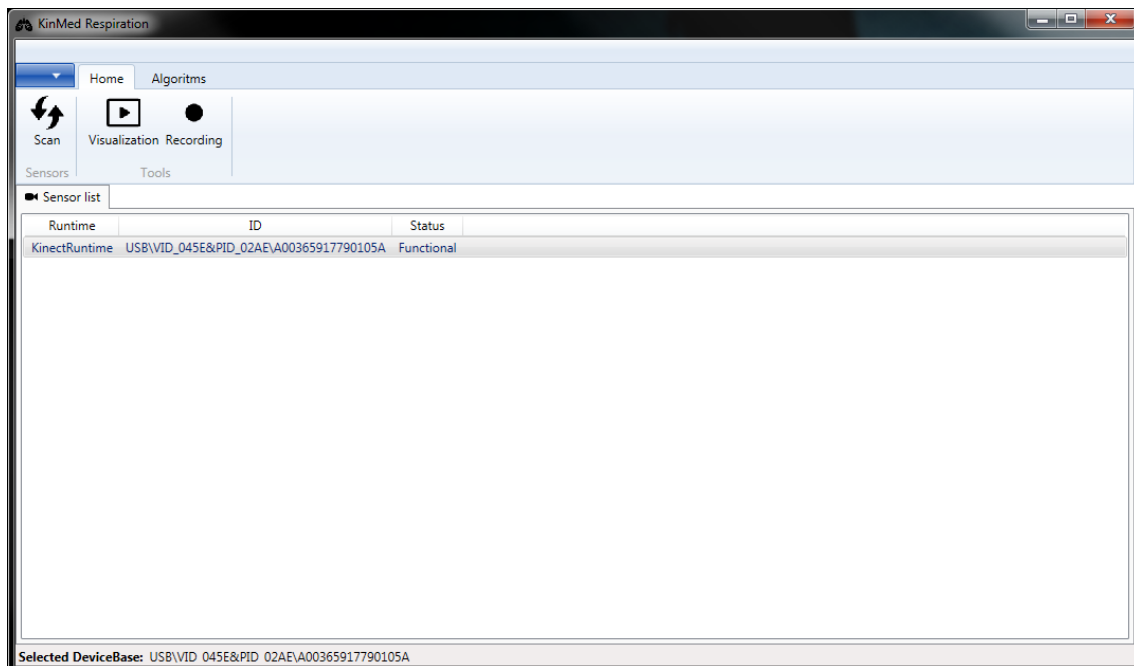


Figura 3.10: Tela principal da interface gráfica do *KinMed*.

separação lógica/GUI e torna a aplicação independente de uma biblioteca gráfica específica. O módulo é dividido nas seguintes partes:

### **KinMed.GUI**

A aplicação em si, o executável que pode ser rodado pelo usuário. É neste componente que todos os outros módulos são conectados. A GUI constitui-se em uma janela principal onde é possível selecionar um sensor e então escolher uma das opções de uso geral, como gravação de dados ou visualização de suas imagens, ou conectá-lo a um dos algoritmos instalados, como pode ser visto na Figura 3.10.

Ocupando a maior área, há uma lista onde todos os sensores ativos e seus respectivos *Runtimes* são enumerados. Ao duplo clique, uma nova aba é criada onde é possível alterar as configurações do sensor, como pode ser visto na Figura 3.11. As configurações disponíveis são dependentes do *SDK* do sensor utilizado, logo a nova aba não é definida na aplicação principal, mas sim no módulo do *Runtime*. Na figura está representada a aba de configuração do sensor *Kinect*®), onde é possível mudar configurações de cada tipo de *stream*, desligar e ligar o sensor infravermelho e mudar o ângulo do sensor.

### **KinMed.RegionDiscover.GUI**

Após a seleção de um sensor, ao se clicar na aba “*Algorithms*”, é possível criar instâncias de todos os algoritmos de módulos instalados e iniciados. Na Figura 3.12 é apresentada uma imagem da tela mostrando uma instância do módulo

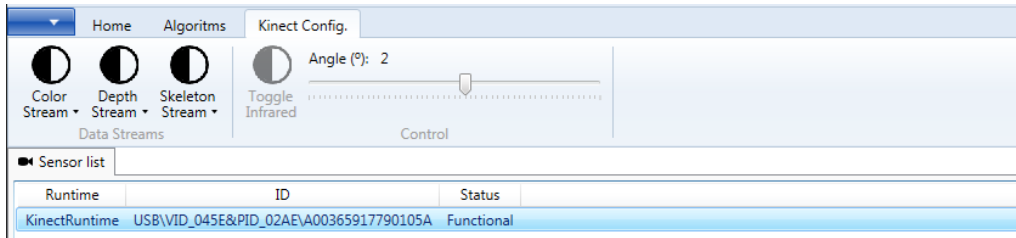


Figura 3.11: Aba de configuração do sensor *Kinect*<sup>®</sup> no *KinMed*.

*KinMed.RegionDiscover* em funcionamento com sua interface própria. Similarmente à aba de configuração de sensores, esta GUI é definida pelo desenvolvedor do módulo de algoritmo que é carregada ao ser criada uma nova instância.

A interface de visualização constitui-se de três componentes principais:

**a Janela de visualização da região torácica e abdominal:**

Permite a visualização da região torácica e abdominal detectada pelo algoritmo *KinMed.RegionDiscover*. A região se move juntamente com o esqueleto detectado do sujeito e a profundidade de cada ponto em seu interior é apresentada através de um “mapa de calor”, ou seja, profundidades mais próximas do sensor são representadas por cores mais quentes (vermelho) e profundidades mais distantes por cores mais frias (azul). Isso permite ao usuário uma análise visual qualitativa de como o tórax está se deformando.

**b Gráfico da profundidade média da região torácica versus tempo:**

Neste gráfico é plotado, em tempo real, o valor da profundidade média calculada pelo algoritmo versus tempo desde sua ativação.

**c Painel de opções do algoritmo:**

Este painel, quando expandido, apresenta várias configurações que vão desde a ativação do algoritmo até congelar a região torácica em um ponto fixo, atribuir um *off-set* a posição dos vértices da região, alterar a visualização dos gráficos e gravar os dados coletados em arquivos externos.

### **KinMed.RespirationPeriod.GUI**

Similarmente à janela anterior, o algoritmo *KinMed.RespirationPeriod* possui uma aba própria para configuração e visualização de dados. Esta aba permite o carregamento dos dados de profundidade média gravados anteriormente pelo *algoritmo de detecção de região torácica* e seu processamento pelo *algoritmo de cálculo de período respiratório*, além da modificação de parâmetros do mesmo e exportação de dados.

Uma tela da interface é mostrada na Figura 3.13. Ela consiste de seis componentes que serão explicados em seguida. Como observação, todos os gráficos são interativos, permitindo ampliação e visualização de dados.

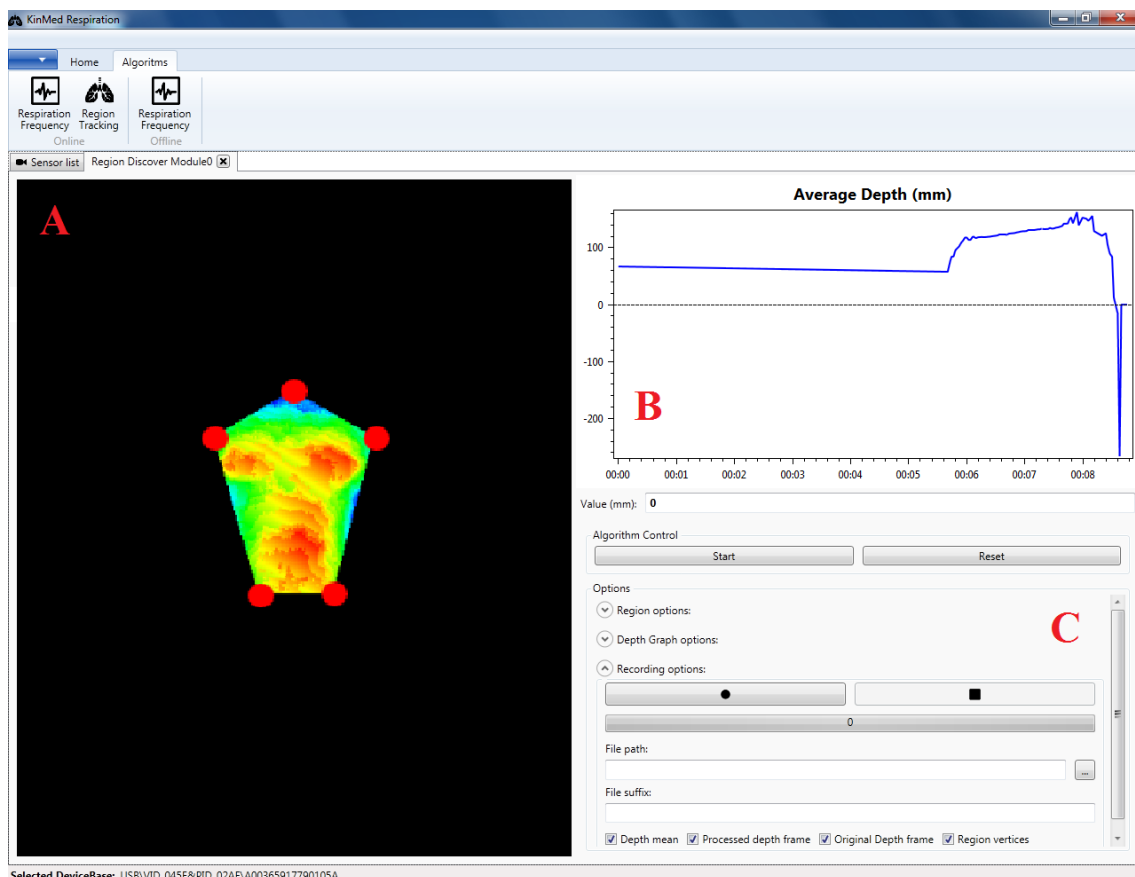


Figura 3.12: Tela da GUI do algoritmo *KinMed.RegionDiscover*. (a) Visualização da região detectada. (b) Gráfico da profundidade média da região. (c) Painel de controle do algoritmo.

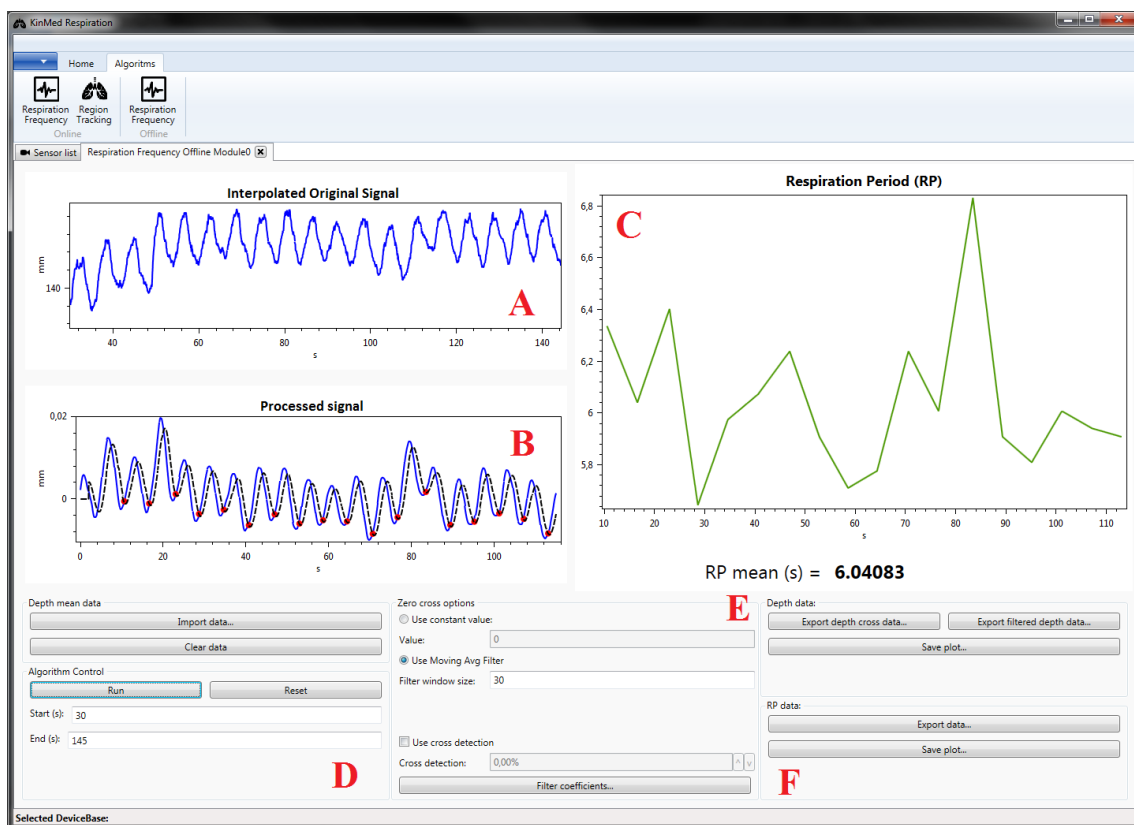


Figura 3.13: Tela do algoritmo *KinMed.RespirationPeriod*. a) Gráfico de profundidade média interpolado. b) Gráfico da derivada da profundidade média. c) Gráfico do período respiratório. d) Importação de dados e controle do algoritmo. e) Configuração do algoritmo. f) Exportação de dados.

#### **a Gráfico de profundidade média interpolado**

Assim que dados de profundidade média são lidos de disco, o algoritmo automaticamente interpola estes dados para uma taxa de amostragem de 30 Hz e então plota este gráfico. Sua funcionalidade principal é a checagem dos dados gravados e a determinação dos tempos de começo e término da análise.

#### **b Gráfico da derivada da profundidade média**

Após rodar o algoritmo pela primeira vez, a derivada da profundidade média é mostrada num gráfico juntamente com a linha pontilhada da média móvel utilizada para a detecção dos ciclos e pontos vermelhos indicando o início de cada ciclo.

#### **c Gráfico do período respiratório**

Ao final da execução do algoritmo, este gráfico é plotado. Ele representa a variação do período respiratório ao longo do tempo de gravação. Abaixo do gráfico, o valor médio do mesmo é mostrado.

#### **d Importação de dados e controle do algoritmo**

No painel superior há botões para importar e limpar dados externos em .csv. Abaixo existe uma seção para o controle da execução (início) e redefinição (para valores padrão) dos parâmetros do algoritmo. Também é possível, pelas duas caixas mais abaixo, definir a faixa no tempo (início e fim) em que o período deve ser calculado.

#### **e Configuração do algoritmo**

Nesta seção, é possível o ajuste de alguns parâmetros e dos coeficientes usados no filtro de 2<sup>a</sup> ordem para filtragem do sinal de profundidade média.

#### **f Exportação de dados**

Por último, este painel possui botões para exportação dos dados processados e de imagens dos gráficos plotados.

### **3.3.6 Visão geral da arquitetura e fluxo de dados**

Com todos os módulos do *KinMed* descritos, é possível montar um diagrama com uma visão geral do programa, ilustrado na Figura 3.14. Nela o comportamento modular é representado através das conexões, dos módulos de Adaptadores, Algoritmos e GUI. Este funcionamento permite diferentes desenvolvedores programarem novos módulos com novas funcionalidades para o *KinMed* sem que seja necessário a alteração dos outros componentes.

Um exemplo seria a adição de um novo algoritmo, digamos, para monitorização postural. Basta o desenvolvedor registrar o novo módulo dele no gerenciador correto,

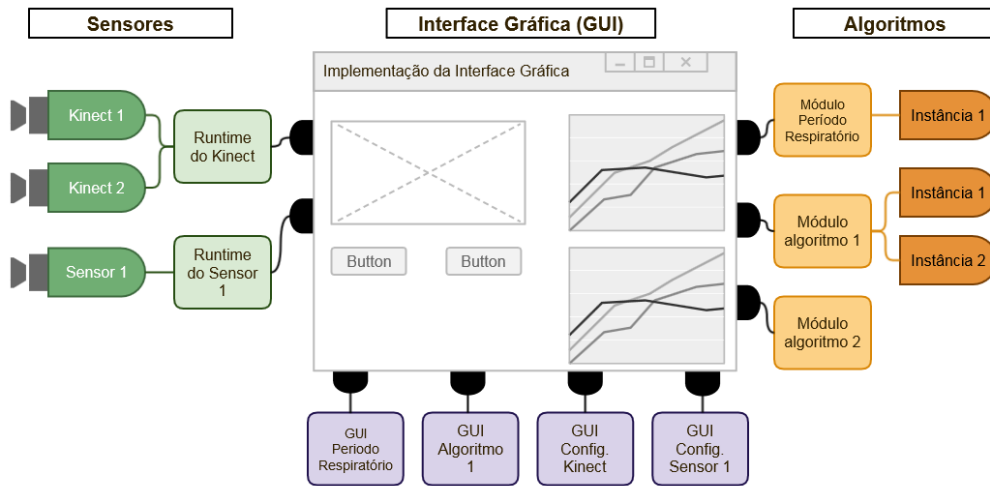


Figura 3.14: Diagrama geral da implementação do *KinMed*, incluindo alguns módulos.

criar uma interface gráfica caso necessário e requisitar ao gerenciador de dispositivos um sensor compatível com os dados que ele precisa; desse modo, o programador não precisa referenciar e nem utilizar a *SDK* do sensor. Com esta funcionalidade, cumprem-se os objetivos 2 e 3 da arquitetura proposta no início desta seção.

# Capítulo 4

## Materiais e Métodos

### 4.1 Sensor *Kinect*®

O sensor *Kinect*® utilizado neste trabalho é o modelo para o console *XBOX*®360 (*Microsoft*®, EUA) ligado a um computador (PC) pelo próprio cabo USB 2.0 do sensor. O PC utilizado é um computador de mesa comum, contendo o programa desenvolvido neste projeto (*KinMed*), o *driver* do *Kinect*® e o *Kinect SDK* 1.8 oficial (ambos desenvolvidos pela *Microsoft*®) instalados. O computador possui a seguinte configuração: Processador Intel Core 2 Quad (2,4GHz), 8 GB de memória RAM e sistema operacional *Windows*®7 SP1.

### 4.2 Pneumotacógrafo

O pneumotacógrafo (PT) permite a medição de vazão de ar e é considerado um equipamento padrão para coleta desse tipo de dado. O PT utilizado nesse trabalho é de orifício variável (modelo 279331, Hamilton Medical AG, Suíça) ligado a um transdutor diferencial de pressão (modelo 279331, Hamilton Medical AG, Suíça). O transdutor condiciona o sinal captado, amplificando-o e filtrando-o com um passa baixa de 33 Hz tipo *Butterworth* de quarta ordem. O sinal é enviado para uma placa de aquisição (NI-USB 6009, National Instruments, EUA) ligada por USB a um laptop (Inspiron N4010, DELL, EUA) de 4,0 GB de memória RAM, processador Intel Core i5 (2,67 GHz) e sistema operacional *Windows*®7 SP1

A aquisição dos sinais digitalizados foi realizada com o programa *Data Acquisition System (DAS)* [15] elaborado em *LabView*® (National Instruments, EUA). O programa foi configurado para colher o sinal a uma frequência de amostragem de 200 Hz e ajustar os ganhos de acordo com um polinômio de calibração. Por fim, o sinal foi também gravado em formato binário.



### 4.2.1 Calibração do Pneumotacógrafo

Para realizar a calibração do PT foi utilizada uma seringa própria para calibração com volume ajustável e seguindo a mesma técnica do trabalho de LUSTOSA [6]: um ajuste polinomial utilizando dados de queda de pressão.

Neste trabalho, a seringa foi ajustada para 3 L e a frequência de amostragem foi de 200 Hz, foram realizadas 20 injeções e o polinômio escolhido foi de 3<sup>o</sup> grau. A aquisição do sinal das injeções foi realizada com o programa DAS [15] e o cálculo dos coeficientes, bem como a estimativa do erro de calibração, foi efetuado com as rotinas de calibração do programa *Mecânica* [16].

## 4.3 Metodologia de teste

A fim de comparar o desempenho do *KinMed* com o do pneumotacógrafo, foram realizados testes onde ambos os sinais foram capturados simultaneamente e em seguida, processados da mesma forma, possibilitando a comparação. O teste constituiu-se da medida individual dos períodos respiratórios (PR) de sujeitos apoiados numa parede a frente do *Kinect*®.

Cada sujeito foi posicionado 1,80 m a frente do *Kinect*® e alinhado ao sensor de profundidade do mesmo. O *Kinect*® foi apoiado numa mesa a 1,06 m do chão, com uma inclinação de 0<sup>o</sup> em relação à mesa, possibilitando assim, o campo de visão do sensor incluir quase inteiramente o corpo do sujeito (com exceção do segmento abaixo do joelho). O sensor é ligado num computador *Desktop*, descrito anteriormente, com o *KinMed* rodando. Foi ligada uma máscara facial acoplada a um PT de orifício variável, onde a diferença de pressão é medida. Os dois lados da placa de orifício são conectados por tubos a um módulo com transdutores diferenciais de pressão e placas de aquisição de sinais, conforme a ilustração da Figura 4.1.

O sinal de pressão do PT foi enviado via cabo USB a um laptop e tratado pelo programa *DAS* [15], onde o sinal da diferença de pressão é transformado em um sinal de vazão de ar. Para tal, a linha média do sinal é subtraída e é aplicado o polinômio de calibração calculado anteriormente.

Alguns cuidados com o ambiente devem ser tomados, por exemplo, os tubos que ligam a placa de orifício e o módulo de aquisição não devem estar no campo de visão (na região calculada pelo *KinMed*) do sensor; Caso contrário, a distância destes ao sensor seria adicionada à profundidade média, causando erros de medida. Outro cuidado consistiu no uso de dois computadores distintos para a aquisição dos sinais, evitando-se, assim, possíveis travamentos ou disputa pelo processamento da *CPU*. Além disso, o ambiente deve estar livre de distúrbios sonoros e de interferência por luz infravermelha.

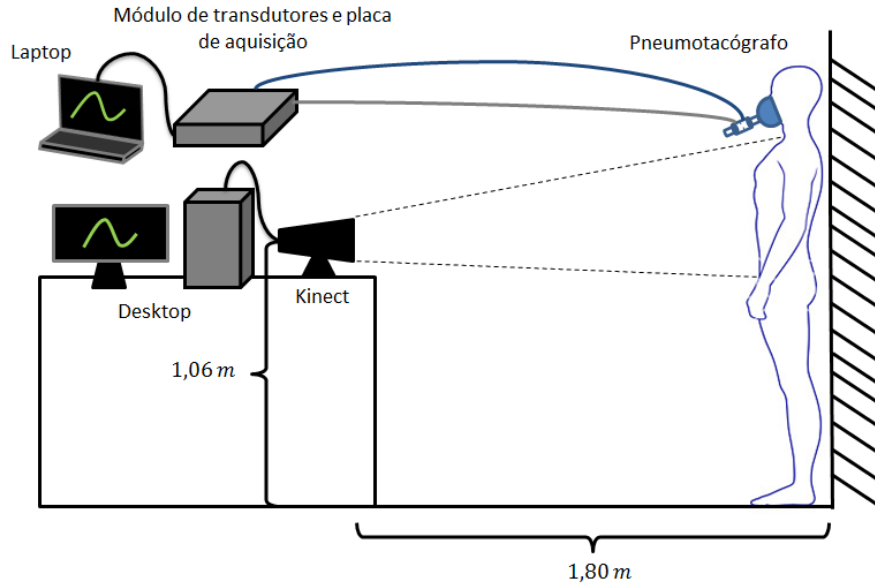


Figura 4.1: Montagem experimental do teste comparativo.

O teste foi dividido em duas partes: na primeira, chamada de respiração espontânea, o sujeito respirou livremente por 2 minutos. Na segunda, com o nome de respiração controlada, ele foi instruído a controlar seu período respiratório de acordo com o ritmo de um metrônomo ajustado para tocar um sinal sonoro a uma frequência de 10, 15 e 20 vezes por minuto, a cada toque ele trocou de inspiração para expiração ou vice-versa. Estas frequências equivalem a 10, 15 e 20 incursões por minuto (irpm) ou a PR de 6s, 4s e 3s, respectivamente. O teste teve duração de 3 minutos, sendo o primeiro minuto para adaptação ao ritmo, o segundo para aquisição de dados e o último para descanso. A sincronização dos dados foi feita pedindo ao sujeito para realizar uma apneia durante 10 segundos no início e no final do teste, gerando um sinal constante em ambos os sensores. Na Figura 4.2 este procedimento é ilustrado através de um diagrama.

Os mapas de profundidade coletados pelo *Kinect*<sup>®</sup> foram processados pelo *Algoritmo de detecção de região torácica* do *KinMed*, gerando um sinal de profundidade média da região que foi gravado em disco. Simultaneamente, o sinal da vazão de ar captado pelo PT foi gravado em disco pelo *DAS* [15]. Ambos os sinais são processados por uma rotina em *Matlab*<sup>®</sup> e por fim, o sinal do *Kinect*<sup>®</sup> foi processado pelo *Algoritmo de cálculo de período respiratório* implementado no *KinMed* para comparação.

A rotina em *Matlab*<sup>®</sup> foi desenvolvida para processar dados de ambas as fontes (*Kinect*<sup>®</sup> e PT) a fim de compará-los, e para tal ela utiliza a ferramenta *Mecânica* [16]. Ela constitui-se de programas para processamento e cálculo de dados relacionados com a respiração, entre elas, a identificação de ciclos respiratórios. A rotina constitui-se de:

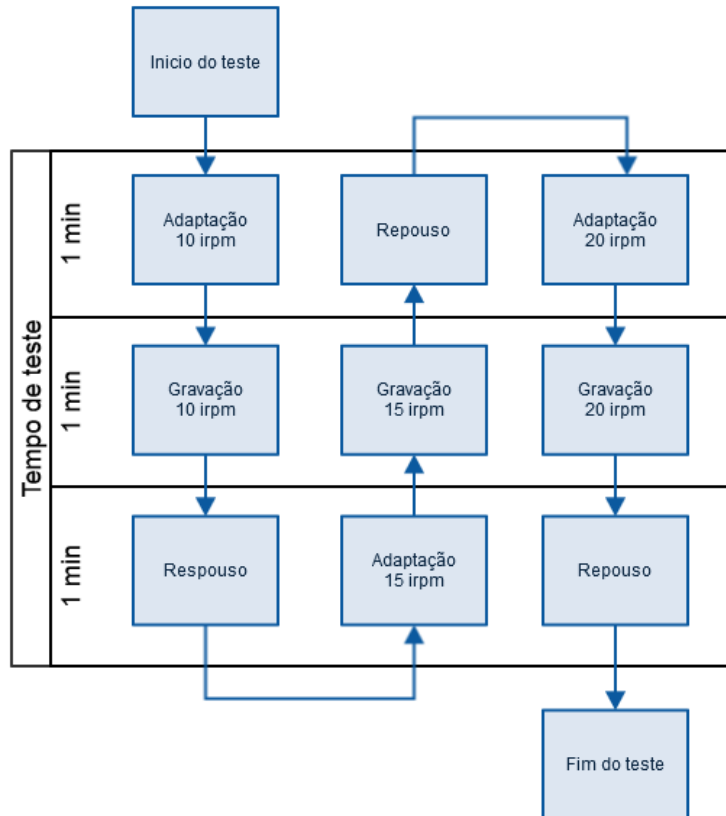


Figura 4.2: Diagrama da segunda parte do procedimento experimental.

- Seleção visual da faixa de dados entre as duas apneias dos dois sinais;
- Interpolação do sinal do *Kinect*<sup>®</sup> para 200 Hz por uma *Spline* de 3<sup>a</sup> ordem e filtragem pelo filtro passa baixas *Butterworth* de 2<sup>a</sup> ordem;
- Derivação do sinal de profundidade média (DPM) do *Kinect*<sup>®</sup> pelo método de diferenças finitas (reta secantes);
- Alinhamento dos sinais pela função *alignsignals* do *Matlab*<sup>®</sup>. A função *alignsignals* utiliza a correlação cruzada para calcular o atraso entre os dois sinais no ponto onde a correlação é máxima, então o sinal mais adiantado é atrasado adicionando-se zeros ao início deste;
- Calibração da amplitude do sinal do *Kinect*<sup>®</sup> por um polinômio de 1<sup>o</sup> grau calculado através de regressão linear da curva (DPM X vazão de ar pelo PT);
- Detecção dos ciclos pelo programa *MECANICA* [16] dois sinais;
- Cálculo dos períodos respiratórios (PR) para cada ciclo e sua média total;
- Cálculo da média dos PR para cada sinal;

- Cálculo do erro relativo absoluto da média dos PRs do *Kinect*<sup>®</sup> em relação a média dos PRs do PT;
- Repetição do último passo acima para os PR obtidos pela execução do algoritmo no *KinMed*;
- Plotagem.

# Capítulo 5

## Resultados

O procedimento experimental descrito anteriormente foi realizado três vezes com os autores deste trabalho (idade: 25 e 43 anos). O ambiente escolhido para o experimento foi iluminado artificialmente por lâmpadas fosforescentes e sem interferência por raios infravermelhos, já que os mesmos poderiam alterar a leitura de profundidade pelo *Kinect*®. A montagem experimental constituiu-se de uma mesa onde se encontra o sensor e o PC a ele conectado rodando o *KinMed*, sendo o sensor sobre um apoio para elevá-lo a 1,06 m do chão. Ao lado desta mesa, encontra-se a placa de aquisição de sinais conectada ao um laptop e a máscara por tubos.

O experimento com respiração espontânea foi realizado individualmente com os dois sujeitos respirando naturalmente durante cerca de 2 minutos. A segunda parte, com respiração controlada, ocorreu em três baterias para cada sujeito. Em ambas as partes, os sinais da vazão de ar e da profundidade média foram coletados respectivamente pelo PT, através da placa de aquisição, e pelo *Kinect*®, através do *KinMed*. A sincronização, tanto do início quanto do final, foi realizada pedindo ao sujeito para realizar uma apneia durante cerca de 10 segundos, gerando um sinal constante que foi identificado visualmente. No total, dois arquivos por bateria e por sujeito foram gerados, totalizando dezesseis arquivos por experimento.

Na Figura 5.1 é possível ver trechos dos sinais de vazão de ar capturados pelo PT e de profundidade média pelo *Kinect*®, ambos sem tratamento.

Ao executar a rotina em *Matlab*®, o sinal do *Kinect*® é filtrado e sua derivada calculada, possibilitando uma calibração entre a variação de profundidade média e a vazão de ar, vista na Figura 5.2. O polinômio escolhido para calibração foi de primeiro grau (reta), por sua simplicidade e pelo fato de apenas alterar a amplitude do sinal. Realizando uma regressão linear entre os pontos da primeira tomada de dados do primeiro experimento, por exemplo, resulta num coeficiente da reta de **0,2318 L/mm**.

A execução dos algoritmos implementados em *Matlab*® e no *KinMed* resulta nos períodos respiratórios médios calculados por: PT através da rotina em *Matlab*

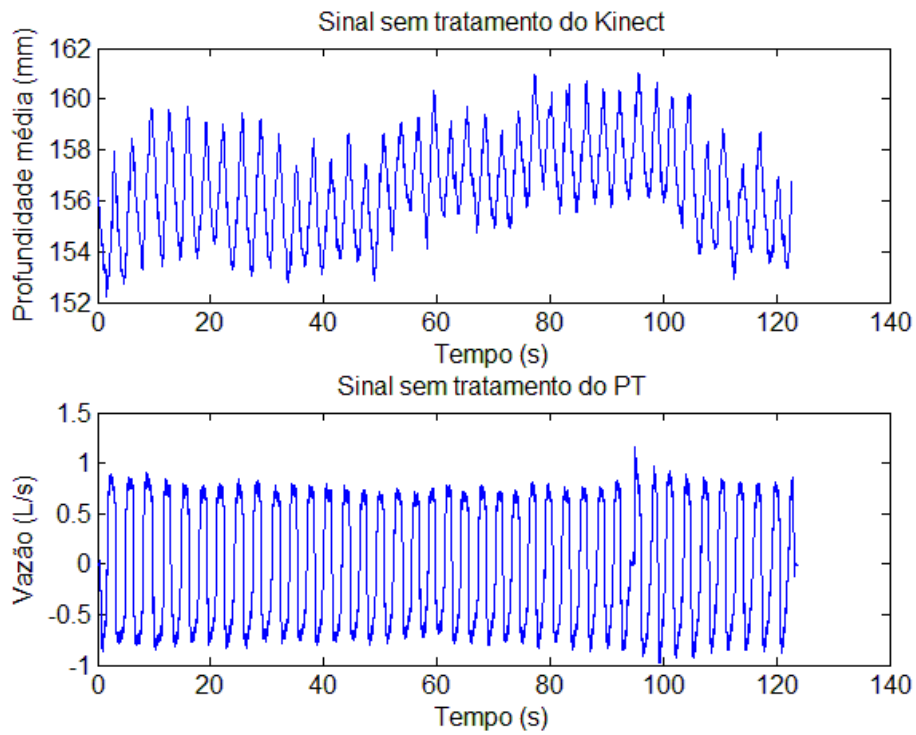


Figura 5.1: Sinais colhidos pelas duas fontes (PT e *Kinect*<sup>®</sup>) antes de tratamento.

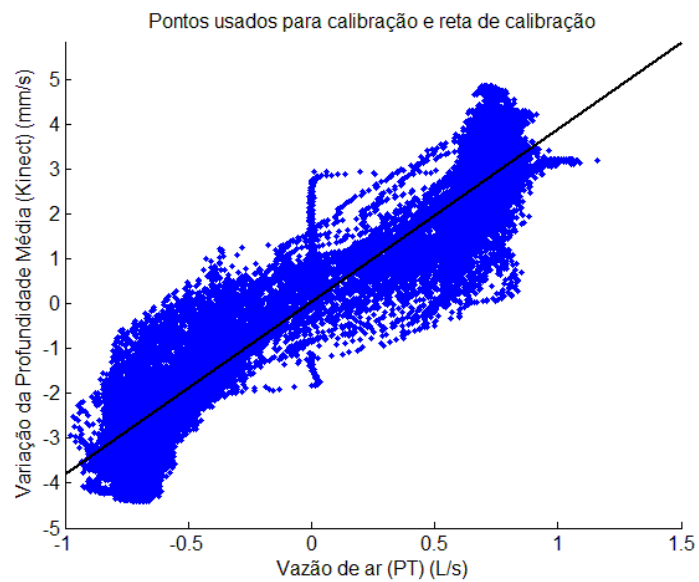


Figura 5.2: Vazão de ar medida pelo PT versus variação da profundidade média medida pelo (*Kinect*<sup>®</sup>) e reta de calibração.

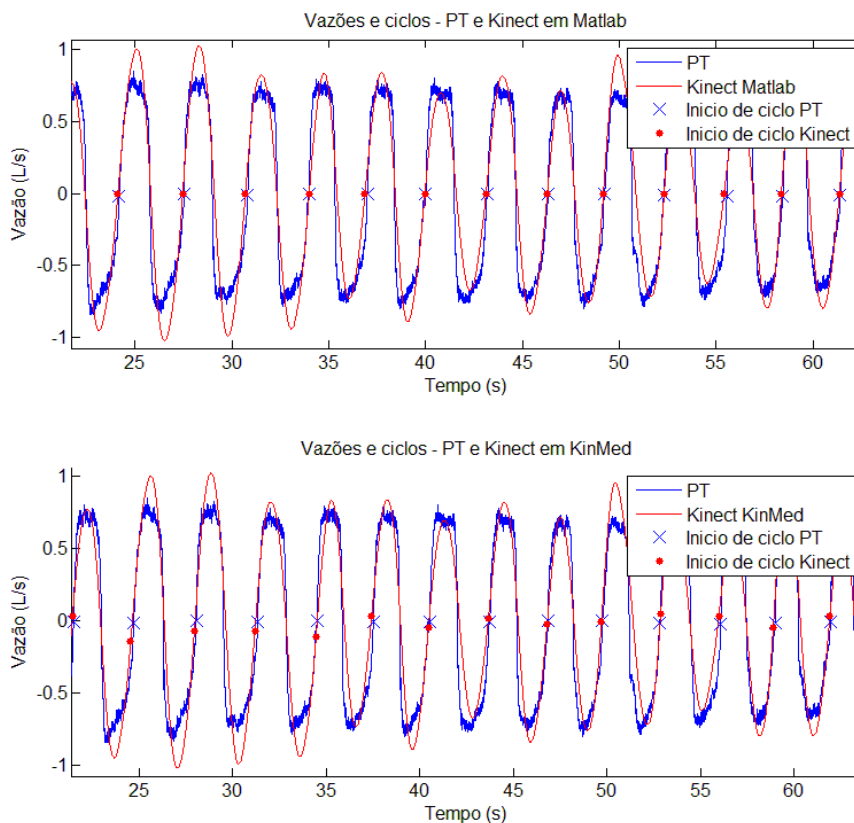


Figura 5.3: Comparação entre os sinais de vazão de ar e profundidade média medidos pelo PT (azul) e pelo *Kinect*<sup>®</sup> (vermelho), respectivamente. Os marcadores representam o início de um ciclo respiratório. O gráfico superior mostra os sinais dos dois sensores quando processados pelo *Matlab*<sup>®</sup>; O gráfico inferior mostra os sinais do PT processado por *Matlab* e o *Kinect*<sup>®</sup> pelo *Kinmed*.

(**PT Matlab**), *Kinect*<sup>®</sup> através da rotina em *Matlab* (**Kinect Matlab**) e *Kinect*<sup>®</sup> através da implementação no *KinMed* (**Kinect KinMed**).

Apesar do sinal sem tratamento do *Kinect*<sup>®</sup> (primeiro gráfico da Figura 5.1) aparentemente não se assemelhar ao sinal de vazão do PT, ao derivá-lo, obtêm-se uma curva muito semelhante, como pode ser visto na Figura 5.3, reforçando a ideia de que a derivada da profundidade média pode ser usada como análogo à vazão. O primeiro gráfico ilustra os sinais de ambos os sensores: "**PT Matlab**" e "**Kinect Matlab**". Os pontos marcados com "x" ou ".", respectivamente para o PT e para o *Kinect*<sup>®</sup>, representam o início de um ciclo detectado pela rotina. No segundo gráfico, são plotados os sinais "**PT Matlab**" e "**Kinect KinMed**" e os marcadores de início de ciclo.

Durante este processamento, foi observado que uma frequência de corte de 0,5 Hz para o filtro *Butterworth* (tanto no *Matlab*<sup>®</sup> quanto no *KinMed*) e uma janela de 40 pontos para o filtro de média móvel (no *KinMed*) apresentaram a melhor detecção

Tabela 5.1: Resultados do teste de respiração espontânea.

Experimento	Sujeito	PR (s)			Erro absoluto relativo ao PT (%)	
		Matlab PT	Matlab <i>Kinect</i> ®	KinMed <i>Kinect</i> ®	Matlab <i>Kinect</i> ®	KinMed <i>Kinect</i> ®
-	-					
I		3,0833	3,0819	3,0800	0,045	0,107
II	I	2,9753	2,9715	2,9770	0,327	0,057
III		2,9485	2,9504	2,9495	0,064	0,034
I		5,9007	5,9043	5,8223	0,061	1,328
II	II	5,0725	5,1021	5,0876	0,583	0,297
III		5,5829	5,5336	5,2130	0,883	6,625
Média:					<b>0,294</b>	<b>1,408</b>
Mediana:					<b>0,096</b>	<b>0,202</b>

de ciclos.

Na Figura 5.4 é possível ver os períodos calculados para cada ciclo detectado pelos três processos. Como se pode observar, os sinais capturados pelo *Kinect*® em ambos os métodos tem forma similar ao padrão ouro e quando comparadas as duas implementações (*Matlab*® e no *KinMed*) são quase idênticos.

Uma forma alternativa de comparação é pelo uso do gráfico de Bland–Altman [17], na Figura 5.5, o gráfico de Bland–Altman foi plotado para a segunda parte do experimento para cada um dos sujeitos. As três cores (azul, vermelho e preto) representam experimentos diferentes, enquanto que as formas dos pontos indicam uma frequência diferente: "x" para 20 irpm, losangos para 15 irpm e "o" para 10 irpm. A linha azul representa a média das diferenças entre pares de amostras e as vermelhas representam o valor da média  $\pm 196\%$  do desvio padrão.

Foram calculados os erros absolutos do "**Kinect Matlab**" e do "**Kinect KinMed**" relativos ao "**PT Matlab**" (o padrão ouro), para fins de comparação. Os resultados para o teste de respiração espontânea podem ser vistos na Tabela 5.1, enquanto que os de respiração controlada podem ser vistos na Tabela 5.2, apresentados aqui no formato "*Mediana {Min:Max}*" dos três experimentos.



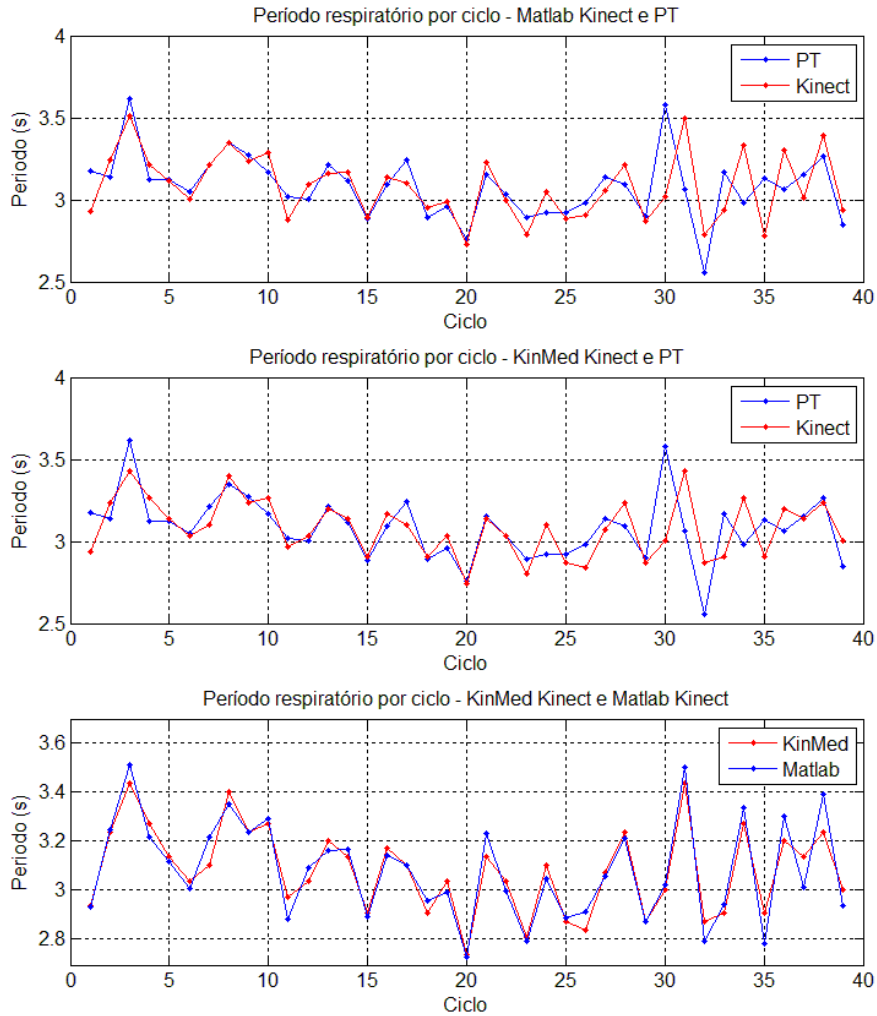


Figura 5.4: Períodos calculados pela rotina em *Matlab*<sup>®</sup>, pelo *KinMed* e seus respectivos sensores. Gráfico superior mostra o período calculado ao longo do tempo medido pelo PT (azul) e *Kinect* (vermelho) na rotina em *Matlab*; Gráfico intermediário mostra o período calculado ao longo do tempo medido pelo PT (azul) na rotina em *Matlab* e pelo *Kinect* (vermelho) no programa *KinMed*; Gráfico inferior mostra o período calculado ao longo do tempo medido pelo *Kinect* na rotina em *Matlab* (azul) e no programa *KinMed* (vermelho).

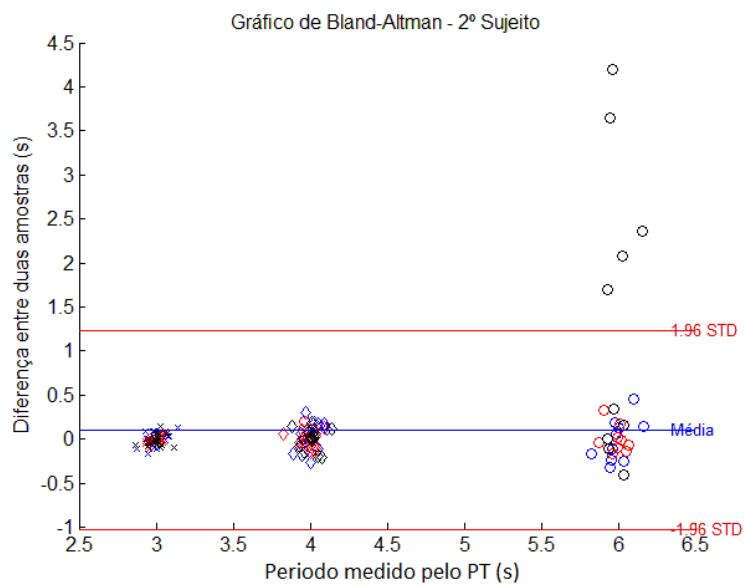
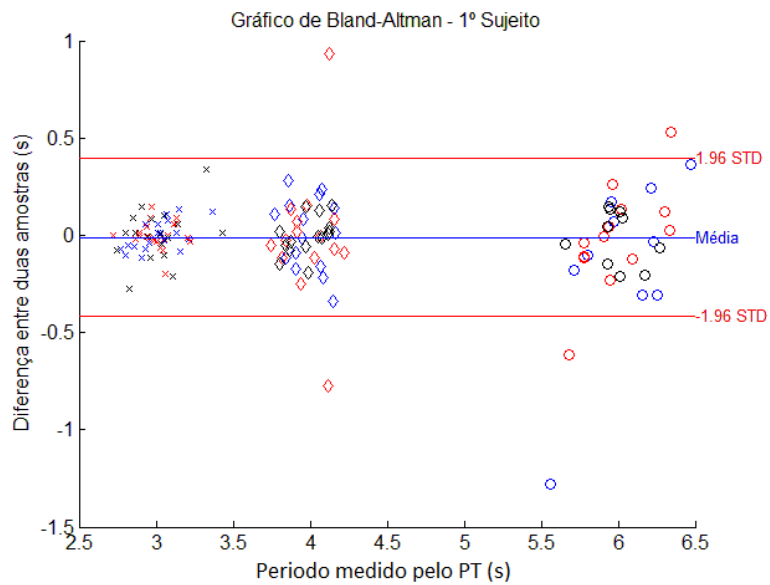


Figura 5.5: Gráfico de Bland-Altman para os três experimentos. As três cores (azul, vermelho e preto) representam experimentos diferentes, enquanto que as formas dos pontos indicam uma frequência diferente: "x" para 20 irpm, losangos para 15 irpm e "o" para 10 irpm. A linha azul representa a média das diferenças entre pares de amostras e as vermelhas representam o valor da média  $\pm 1,96\%$  do desvio padrão.

Tabela 5.2: Resultados do teste de respiração controlada, apresentado aqui da forma "Mediana {Min:Max}"

PR esperado (s)	Sujeito	PR (s)			Erro absoluto relativo ao esperado (%)		
		Matlab PT	Matlab Kinect	KinMed Kinect	Matlab PT	Matlab Kinect	KinMed Kinect
-	-	3,004	3,006	3,005	0,323	0,490	0,160
3,0	I	{2,985:3,010}	{2,985:3,032}	{2,998:3,011}	{0,120:0,500}	{0,197:1,077}	{0,060:0,375}
	II	2,999	3,002	2,996	0,093	0,067	0,130
4,0	I	{2,997:3,004}	{2,998:3,008}	{2,995:3,001}	{0,050:0,140}	{0,067:0,267}	{0,043:0,175}
	II	3,987	3,981	3,984	0,318	0,475	0,395
6,0	I	{3,976:3,988}	{3,973:3,993}	{3,971:3,995}	{0,292:0,592}	{0,167:0,680}	{0,123:0,725}
	II	4,004	3,990	3,986	0,168	0,257	0,340
Média:	I	{3,991:4,007}	{3,982:4,005}	{3,962:3,988}	{0,100:0,233}	{0,125:0,450}	{0,295:0,945}
	II	5,991	6,005	5,986	0,550	2,817	0,230
Mediana:	I	{5,920:6,033}	{5,733:6,169}	{5,881:5,996}	{0,150:1,340}	{0,083:4,450}	{0,070:1,990}
	II	5,995	5,763	5,808	0,083	3,950	3,200
		{5,786:5,997}	{3,938:5,884}	{3,318:5,884}	{0,050:3,567}	{3,068:34,375}	{1,935:44,700}
					<b>0,256</b>	<b>1,343</b>	<b>0,742</b>
					<b>0,243</b>	<b>0,483</b>	<b>0,285</b>

# Capítulo 6

## Discussão

### 6.1 Análise da detecção de ciclos respiratórios e seus períodos

Os resultados apresentados na seção anterior mostram a similaridade do desempenho do *Kinect*® ao padrão ouro, o pneumotacógrafo. Analisando o período médio durante a respiração espontânea, os erros absolutos da implementação em *Matlab*® relativos ao padrão ouro têm mediana na ordem de 0,1% não passando de 0,88%; enquanto que o *KinMed* atingiu 0,2% de mediana, porém com erros de até 6,6%. No teste com respiração controlada, o PT apresentou um erro relativo ao valor esperado com mediana de 0,26%; enquanto que o *KinMed* apresentou 0,37% de mediana. Porém, em uma das medidas, a captura de dados apresentou ruídos consideráveis, resultando em erros de até 44,7%, em decorrência de características e comportamentos do *Kinect*® que serão explicados mais a frente.

Na Figura 5.4, correspondente à primeira parte do experimento (respiração espontânea), verifica-se que os períodos calculados pelos três métodos (PT *Matlab*, *Kinect*®*Matlab* e *Kinect*®*Kinmed*) são similares em relação ao tempo e esse comportamento se repete também na segunda parte. Conforme visto no gráfico de Bland-Altman, a qualidade do *Kinect*® em detectar ciclos e períodos respiratórios fica mais clara. A linha média das diferenças tem valor bem próximo de zero e a grande maioria dos pontos se concentra dentro da faixa definida pela média  $\pm 96\%$  do desvio padrão. Mesmo considerando-se o experimento no qual a medição foi problemática, ilustrada pelo segundo gráfico da Figura 5.5, as diferenças de medidas entre os dois sensores é pequena, ela é menor que 1% na maioria dos casos.

Se comparado os desempenhos da rotina em *Matlab*® com a da implementação em C# do *KinMed*, nota-se que ambos tem erros similares e curvas compatíveis em relação às do PT. Na primeira parte do experimento, a diferença no erro absoluto entre os dois métodos não passou de 1,5%, enquanto que na segunda parte os dois

têm praticamente o mesmo desempenho quando ao valor esperado. Esses fatores sugerem que o *algoritmo de cálculo de período respiratório* implementado no *KinMed* tem desempenho compatível com o do programa *Mecânica* quanto à detecção de ciclos respiratórios no sinal de variação de profundidade média do *Kinect®*.

Os estudos citados na Introdução deste trabalho propuseram diversas formas de captura da cinemática torácica através da utilização do *Kinect®*. Similarmente ao estudo de AOKI *et al.* [5], este trabalho utilizou o esqueleto fornecido pelo *Kinect®* para definição da região torácica, automatizando o processo de rastreamento e eliminando *pixels* problemáticos em bordas de superfícies. Porém, uma análise desta região é necessária para determinar se esta engloba as áreas mais responsáveis pela variação do volume durante a respiração.

Mesmo o sinal de profundidade média do *Kinect®* possuindo ruídos consideráveis, os ciclos respiratórios puderam ser detectados em todos os sujeitos e frequências respiratórias do experimento. Uma redução deste ruído talvez possa ser alcançada diminuindo-se a distância do sujeito ao sensor, já que ANDERSEN *et al.* [9] mostraram que a resolução do *Kinect®* diminui com o aumento da profundidade. Entretanto, como o ângulo de visão do sensor é fixo, existe um troca entre resolução e campo de visão, ou seja, uma aproximação do sujeito ao sensor poderia impossibilitar o rastreamento do esqueleto usado para determinar a região torácica ou entrar na distância mínima do sensor IV.

Em todos os estudos, foi mostrado que o sensor tem capacidade para realizar esse tipo de medida com certa exatidão quando comparado ao padrão ouro, o PT. Contudo nota-se que as leituras estão sujeitas a considerável variabilidade e a um ruído de alta frequência. De fato, no estudo conduzido por ANDERSEN *et al.* [9] mostrou-se que pelo menos 20% dos *pixels* em um mesmo plano de profundidade podem assumir três valores diferentes. Porém, deve-se considerar que o sensor foi desenvolvido primariamente para rastreamento de pessoas e seus movimentos, sendo estes normalmente de alta amplitude e de partes do corpo como membros ou tronco, não sendo necessária medida de profundidade com alta resolução.

## 6.2 Problemas e limitações

### 6.2.1 Limitações do *Kinect®*

Um experimento complementar foi realizado para testar os limites de resolução do sensor quanto à amplitude do movimento torácico (consequentemente, volume respiratório) e à frequência respiratória. Utilizando-se as mesmas condições do experimento principal, um sujeito foi instruído a para respirar espontaneamente, depois realizar ciclos respiratórios com grande movimentação torácica e com períodos lon-

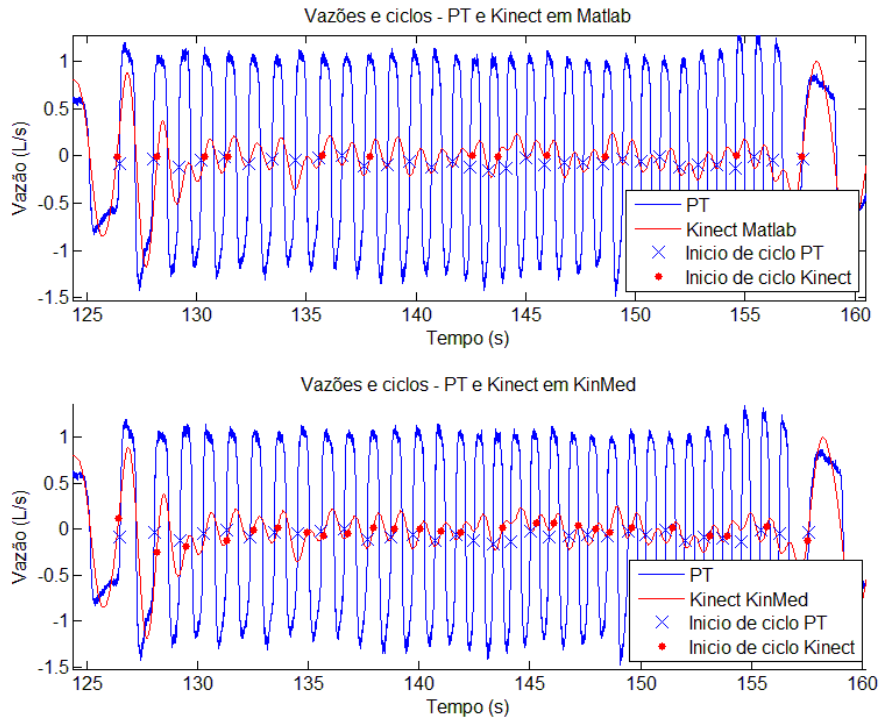


Figura 6.1: Trecho da gravação de vazão de ar e a variação de profundidade média no teste complementar de resolução, onde o sujeito respirou em alta frequência e baixa movimentação torácica. Gráfico superior mostra um trecho gravado com o PT e o *Kinect*<sup>®</sup> e processado pelo *Matlab*<sup>®</sup>; Gráfico inferior mostra um trecho gravado com o PT, processado pelo *Matlab*<sup>®</sup>, e com o *Kinect*<sup>®</sup>, processado pelo *KinMed*

gos e em seguida, respirações curtas e com baixa movimentação torácica. Os sinais de profundidade média medidos pelo *Kinect*<sup>®</sup> e de vazão de ar coletado pelo PT foram processadas pela rotina em *Matlab*<sup>®</sup> e também pelo *KinMed*.

No trecho com períodos longos e amplitudes grandes, o sensor se comportou similarmente ao experimento, conforme esperado. Na Figura 6.1 é possível observar um trecho desta gravação para os ciclos com períodos curtos. Nota-se que o *Kinect*<sup>®</sup> não foi capaz de medir um sinal similar à vazão para movimentos torácicos pequenos, resultando numa forma de onda com amplitude muito menor, porém com frequência compatível. Se comparados os inícios de ciclos detectados pelos dois sensores, mostrados no primeiro gráfico da Figura 6.1, verifica-se uma consequência negativa deste comportamento: vários ciclos não foram detectados pelo programa *Mecânica* [16] no sinal do *Kinect*<sup>®</sup>, enquanto que no mesmo trecho foram encontrados mais ciclos detectados pelo sinal do PT. Contudo, no segundo gráfico da mesma figura, o *Algoritmo de cálculo de período respiratório* detectou mais ciclos do que o algoritmo do *Mecânica* [16] para o mesmo sinal do *Kinect*<sup>®</sup>. De fato, se forem analisados os três gráficos de período respiratório por tempo calculados pelos dois algoritmos, da

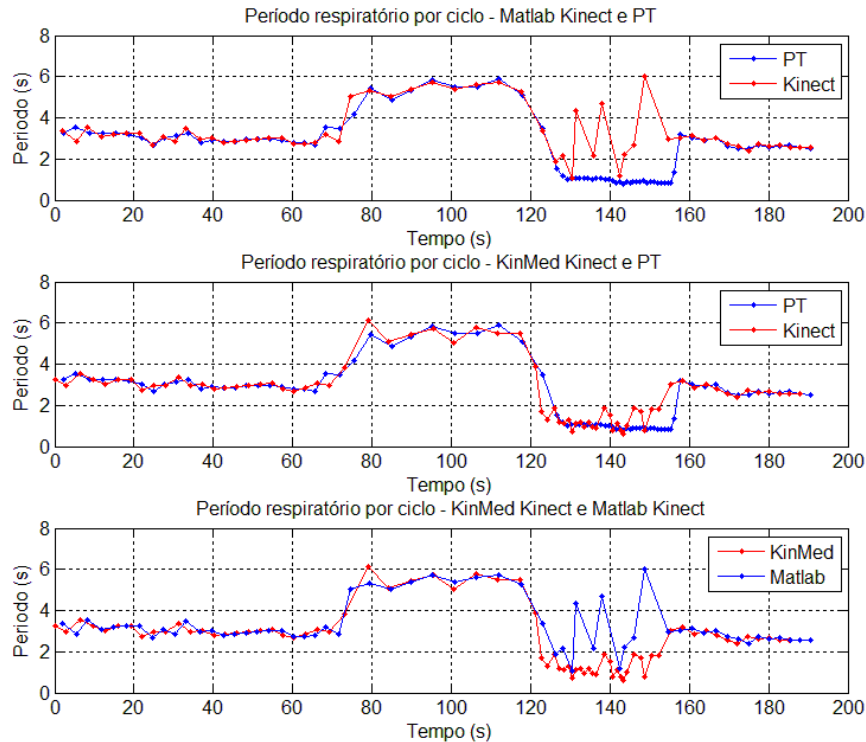
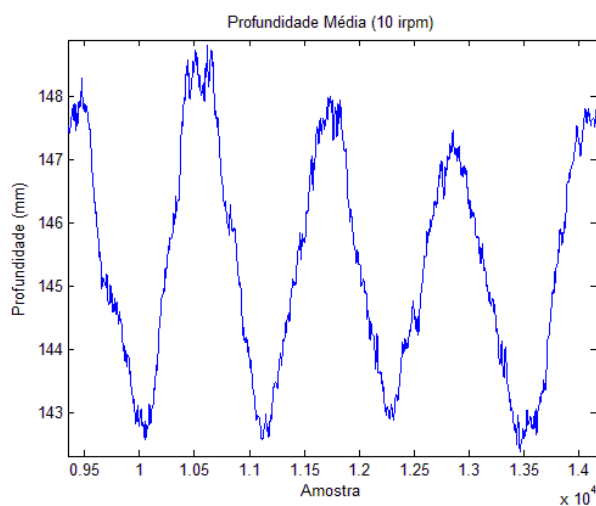


Figura 6.2: Período respiratório por tempo calculados por: (a) *Matlab*<sup>®</sup> *Kinect*<sup>®</sup> e PT. (b) *Kinmed Kinect*<sup>®</sup> e PT. (c) Comparação entre os dois algoritmos.

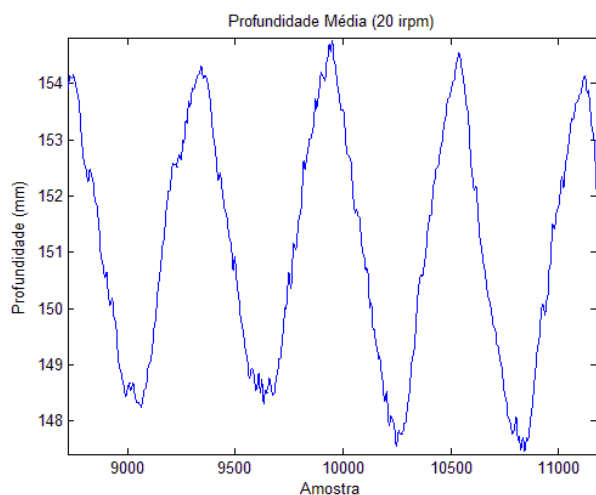
Figura 6.2, pode-se verificar que o *KinMed* obteve valores mais próximos ao do PT para este trecho em questão.

É possível que as diferenças de amplitude nos sinais sejam causadas pela relação movimento torácico e vazão de ar, onde a expansão e contração da caixa torácica tem menos participação no volume inspirado ou expirado para este ritmo/tipo de respiração. Um indicativo desta ideia pode ser visto nos gráficos da Figura 6.1, onde a vazão de ar medida pelo PT permanece com uma amplitude similar às anteriores e posteriores a este trecho, enquanto que o movimento torácico medido pelo *Kinect*<sup>®</sup> sofre uma queda considerável neste período. A diferença no desempenho de detecção entre os dois algoritmos pode ser explicada pelo fato do programa *Mecânica* utilizar um limiar de vazão para identificar uma expiração/inspiração. Conseqüentemente, nem todos os picos do sinal do *Kinect*<sup>®</sup> atingem esse limiar e são descartados. Mais análises da relação movimento torácico e vazão são necessárias, assim como testes com novos algoritmos ou ajustes nos já existentes.

Durante a parte de respiração controlada do experimento, também foi analisada a influência do ruído no sinal de profundidade média com a variação da frequência respiratória. Ao se observar os gráficos de profundidade média para medições do *Kinect*<sup>®</sup> na respiração controlada a 10 irpm e 20 irpm (Figura 6.3), nota-se que o



(a) Recortes do gráfico de variação de profundidade média medida pelo *Kinect*<sup>®</sup> para respiração a 10 irpm.



(b) Recortes do gráfico de variação de profundidade média medida pelo *Kinect*<sup>®</sup> para respiração a 20 irpm.

Figura 6.3: Recortes do gráfico de profundidade média medida pelo *Kinect*<sup>®</sup>. Observe que para 10 irpm o distúrbio causado pelo ruído é mais presente.

ruído tem uma forte influência na forma do sinal de 10 irpm, causando um distúrbio considerável em seus picos, mesmo depois da filtragem. Quando comparado ao segundo gráfico da Figura 6.3b), fica claro que o ruído tem menos influência em frequências mais altas. A relação entre variabilidade e frequência respiratória foi observada por LUSTOSA [6] em sua dissertação, onde também se discute sua origem. Ele atribui a variabilidade da medição do período respiratório em baixas frequências respiratórias à diferença de velocidade de detecção de mudanças de fase respiratória entre o PT e o *Kinect*<sup>®</sup>, já que o PT se baseia na vazão e o *Kinect*<sup>®</sup>, na deformação da superfície torácica.



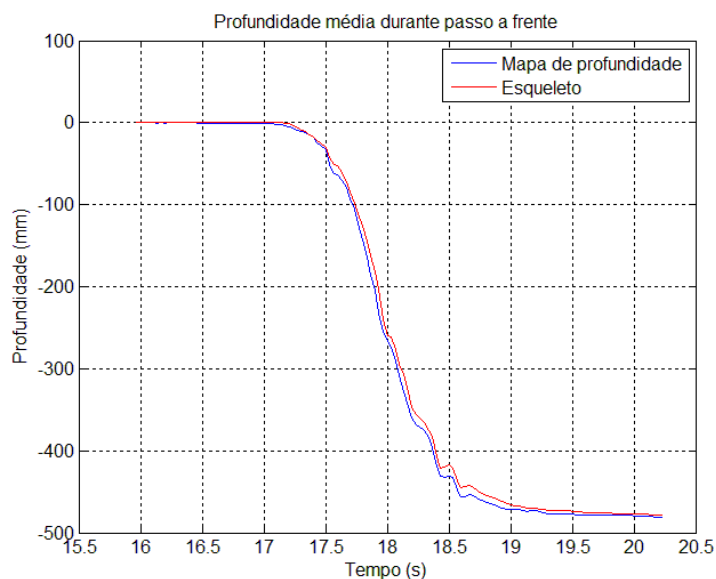


Figura 6.4: Profundidade média do mapa e média dos componentes Z das juntas da região durante o teste de passo. Nota-se que mesmo se tratando de pontos com posições supostamente idênticas, há diferenças tanto na dinâmica quanto na amplitude.

## 6.2.2 Limitações dos algoritmos propostos

Um dos problemas notados no desenvolvimento do *algoritmo de detecção de região torácica* foi a movimentação do esqueleto não acompanhar de imediato a do mapa de profundidade, ou seja, o cálculo das posições das juntas do esqueleto aparenta ser mais lento. Como consequência, o *Algoritmo de detecção de região torácica* não consegue acompanhar perfeitamente o movimento do sujeito, gerando um estado transitório entre o início e fim de um movimento onde há ruídos e picos no sinal de profundidade média, porém, com valor final similar ao do início do movimento. Isso vai contra o comportamento idealizado do sistema plano/esqueleto, onde o valor da profundidade média deveria se manter constante para movimentos e se alterar apenas para deformações da caixa torácica.

Para analisar esta característica, um teste foi realizado, no qual um sujeito se posicionou a cerca de 1,8 m à frente do *Kinect®* e em seguida deu um passo de cerca de 30 cm em direção ao sensor, realizando uma apneia. Os dados de profundidade do sensor foram coletados pelo algoritmo de detecção de região torácica do *KinMed*, além de quadros de profundidade e as posições das cinco juntas utilizadas no cálculo da região e do plano. Os sinais foram tratados em *Matlab®*, calculando-se a profundidade média do mapa e a média da componente Z das juntas do esqueleto alinhados quanto ao valor inicial. O resultado pode ser visto na Figura 6.4.

No gráfico da Figura 6.4 é possível ver diferenças na amplitude dos sinais ao longo do tempo. Como o valor de profundidade de cada pixel na região é calculado pela

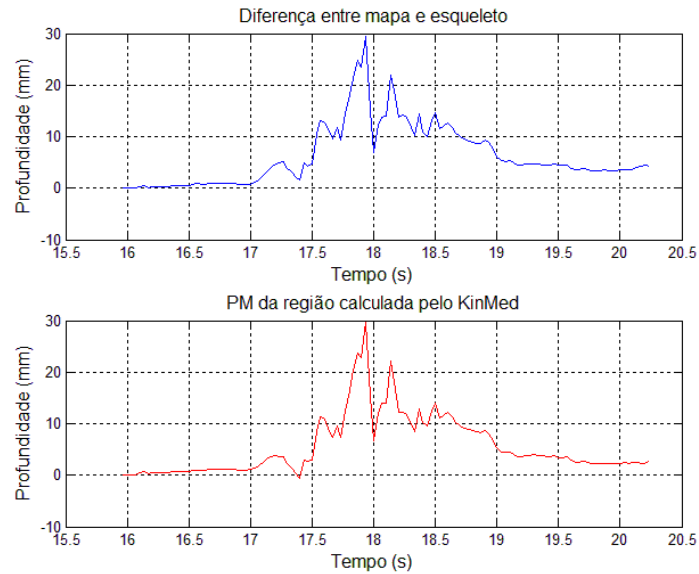


Figura 6.5: Comparativo entre o teste de passo e o sinal calculado pelo *Kinmed*. Acima a diferença entre profundidade média da região e esqueleto, abaixo o sinal calculado pelo algoritmo de detecção de região no *KinMed*. Observe que as duas curvas têm aparência similar.

diferença entre estes dois sinais, em vez da profundidade média assumir um valor idealmente constante, houve picos e ruídos durante a fase de transição do sistema (do início até o final do passo). Realizando a subtração destes dois sinais e os plotando juntamente com o sinal de profundidade média calculado pelo algoritmo, obteve-se ao gráfico da Figura 6.5.

Mesmo pelo fato do resultado da subtração entre os sinais da média da componente Z e da profundidade média do mapa não ser o parâmetro utilizado pelo algoritmo, o sinal resultante tem praticamente a mesma forma, a salvo de um deslocamento. Pode-se verificar que do período de início do movimento em  $t = 17s$  até seu término  $t = 20s$  o comportamento transitório é observado.

A abordagem sugerida no *algoritmo de detecção de região torácica* parece não ser robusta o suficiente para compensar todos os movimentos realizados pela pessoa monitorada, obtendo êxito apenas para pequenos movimentos. Contudo, o método utilizado produz um sinal que tem uma faixa de valores constante independente da posição da pessoa monitorada à frente do sensor, respeitadas suas limitações. Além disso, os cálculos realizados são rápidos o suficiente para um computador moderado executar o algoritmo quadro a quadro. Outra potencial vantagem do método utilizado é o fato de que a profundidade de cada pixel dentro da região é calculada. Essa informação pode, então, ser utilizada para calcular o volume ou analisar áreas com irregularidades durante o ciclo respiratório, por exemplo. Esta limitação fez com que, nos testes os sujeitos fossem instruídos a se apoiar

numa parede, evitando assim movimentos bruscos. Há sugestões para solucionar este problema, algumas ideias são apresentadas abaixo:

- Utilizar o esqueleto provido pelo *Kinect*® apenas como referência, calculando-se o plano de referência a partir da média das profundidades dos *pixels* ao redor das juntas do esqueleto;
- Filtrar o sinal do esqueleto em tempo real;
- Encaixe de modelo tridimensional da caixa torácica no mapa de profundidade;

### 6.2.3 Problemas com o *SDK/Driver*

Durante os experimentos, o sensor apresentou lentidões temporárias na captura de dados, comprometendo a taxa de amostragem nominal constante de 30 Hz (30 FPS). Em algumas ocasiões, o *SDK* parou de fornecer dados durante 2 segundos e após a pausa, os dados chegaram em batelada, prejudicando a aquisição em tempo real. Durante um teste, o *Kinect*® se desligou e parou de fornecer dados. Situações similares ocorreram apenas com seu sensor infravermelho e foi necessário religar o *Kinect*®.

Observou-se que dependendo da inicialização do sensor (acontecendo aleatoriamente), ele apresenta uma captura de quadros de profundidade com um ruído elevado, e religá-lo não resolveu o problema muitas vezes. Isso é ilustrado no 3º experimento com o sujeito II para respiração controlada a 10 irpm, onde este comportamento resultou em erros da ordem de 30% do valor esperado. Curiosamente, ao iniciar a próxima bateria, o sensor voltar a funcionar corretamente, mesmo sem nenhuma mudança no ambiente, nas roupas do sujeito ou no equipamento usado.

É possível que o problema esteja na relação do *driver* ou *SDK* do *Kinect*® com o sistema operacional. Como o comportamento se mostrou aleatório, não foi possível testar adequadamente suas ocorrências. A configuração e desempenho do computador utilizado também podem ter contribuído para estes comportamentos; além do fato de que o *KinMed* precisa ser otimizado e possivelmente melhorado com conceitos de programação paralela.

### 6.2.4 Cuidados experimentais

Ao realizar os experimentos, alguns acontecimentos externos parecem ter afetado a coleta de dados. Em uma ocorrência durante um dos testes, o sujeito se assustou com um barulho alto de fora da sala, produzindo um pico em ambos os sinais. Em outra, os tubos presos na placa de orifício do pneumotacógrafo se soltaram da fita usada para prendê-los na roupa do sujeito, entrando assim no campo de visão do

*Kinect*® e contribuindo negativamente para o cálculo da profundidade média. Outro fator importante foi a escolha da cor e material das roupas utilizadas pelos sujeitos: camisas de cores escuras ou de materiais muito reflexivos dificultam a leitura da profundidade pelo sensor IV, sendo recomendada evitá-las.

### 6.3 Possíveis melhoramentos e trabalhos futuros

Existem possíveis melhoramentos nos algoritmos implementados neste trabalho a ser levados em consideração. A região do *Algoritmo de detecção de região torácica* é definida por cinco juntas do esqueleto; contudo, esta região não inclui toda a superfície torácica abdominal e considera todos os pontos com o mesmo peso. Uma melhor escolha de região e a adição de pesos na contribuição de certos pontos a profundidade média poderia melhorar sua relação com a vazão durante a respiração, além de permitir a medida de outros parâmetros, como a contribuição de certas áreas no volume respiratório.

O *Algoritmo de cálculo de período respiratório* também pode ser melhorado. A interpolação do sinal de profundidade média para uma taxa amostral constante foi realizada com uma *spline* de 3º grau, sendo que outras curvas talvez tragam melhores resultados. A filtragem do sinal deve ser testada quanto ao tipo de filtro e a frequência de corte adotadas. O método escolhido para detecção de ciclos possibilita a transformação desse algoritmo para análise em tempo real. Para tal, algumas modificações devem ser feitas: a interpolação do sinal de profundidade média deve ser em tempo real ou em bateladas e a filtragem deve ser igualmente modificada.

Neste trabalho foram desenvolvidos algoritmos para detecção de ciclos respiratórios e seus períodos, o *Algoritmo de detecção de região torácica* e o *Algoritmo de cálculo de período respiratório*. Contudo, esses algoritmos e a capacidade de gerar mapas de profundidade do *Kinect*® podem ser utilizadas para outras aplicações na área médica. A região calculada pelo *Algoritmo de detecção de região torácica* contém as distâncias de cada pixel do mapa de profundidade ao plano definido pelo esqueleto. Essa informação pode ser usada para reconstruir a superfície torácica em um modelo de três dimensões por *ICP (Iterative closest point)*. Isso permitiria o cálculo de diversos parâmetros respiratórios, como o volume de ar corrente por exemplo. Outra aplicação interessante desse tipo de dado é o acompanhamento do balanço torácico/abdominal da respiração, ou seja, se uma região do tórax ou abdômen está se movimentando de forma diferente de outra, fator que pode apontar certas doenças ou a qualidade da recuperação depois uma cirurgia. Atualmente médicos utilizam a visão para obter uma análise qualitativa ou o paciente pode precisar ser movido de seu leito para radiografar, obtendo assim um resultado quantitativo, conseqüentemente submetendo o paciente, muitas vezes em condições frágeis, a ra-

dição.

A ferramenta *KinMed* foi desenvolvida pensando em trabalhos futuros e sua expansão, por isso os conceitos de arquitetura modular e *Framework* foram adotados. O sistema de dados provido pela ferramenta é genérico e pode ser utilizado para monitorização e análise de outros parâmetros fisiológicos além da respiração. O rastreamento de esqueletos fornecido pelo *Kinect*<sup>®</sup> tem possibilidade de ser aplicado à estabilometria ou cinesiologia. Adicionalmente, um módulo de comunicação em rede pode ser desenvolvido, possibilitando o *KinMed* se comunicar com um atuador ou a participar de uma malha de controle.

# Capítulo 7

## Conclusão

Neste trabalho foi avaliado o desempenho do *Microsoft®Kinect®* na monitorização da frequência respiratória sem contato. Este estudo e outros anteriores mostraram que o *Kinect®* parece ter resolução suficiente e poder ser usado para fins de análise respiratória, com a ressalva de uma moderada variabilidade nas medições. Um novo algoritmo para detecção da região torácica abdominal foi proposto, que tenta compensar os movimentos da deformação da superfície dentro de uma região, visando uma medição mais robusta. Juntamente, foi implementado um algoritmo que usa esses dados para calcular o período respiratório.

Testes comparativos com um padrão ouro (PT) foram realizados e foi observado que os algoritmos têm desempenho similar e de boa qualidade. Contudo, os movimentos dos voluntários não foram completamente anulados pelo *Algoritmo de detecção de região torácica*, pois existe uma limitação na velocidade e exatidão do esqueleto fornecido pelo *Kinect®*. Possíveis soluções para este problema são um tratamento mais inteligente das posições das juntas do esqueleto ou a utilização em conjunto do mapa de profundidade e das juntas.

O mapa de profundidade processado pelo *Algoritmo de detecção de região torácica* pode possuir outra função interessante para futuros trabalhos: como cada ponto da superfície dentro da região torácica abdominal tem sua profundidade até o plano definido pelas juntas do esqueleto disponível, eles talvez possam ser usados como pontos de referência para construção de modelos respiratórios, utilizando, por exemplo, *ICP (Iterative closest point)* para encaixe de modelos tridimensionais ou para o cálculo do volume torácico através de uma integração numérica, como sugerido por AOKI *et al.* [5].

Um dos focos neste trabalho foi desenvolver uma ferramenta de software que pode ser usada para futuros trabalhos na área médica envolvendo o *Kinect®* ou sensor similar. Esta ferramenta é independente de outros softwares e tem uma arquitetura modular e expansível, além de possuir interface gráfica própria e amigável. Apesar de neste trabalho ela ser utilizada para medições de propriedades respiratórias, ela

pode ter suas funcionalidades expandidas para englobar análise estilométrica, por exemplo, sem alteração significativa de seu código fonte, graças a sua arquitetura modular e independência do sensor utilizado.

# Referências Bibliográficas

- [1] AL-KHALIDI, F. Q., SAATCHI, R., BURKE, D., et al. “Respiration rate monitoring methods: A review”, *Pediatric pulmonology*, v. 46, n. 6, pp. 523–529, jun. 2011.
- [2] HAN, J. N., STEGEN, K., CAUBERGHS, M., et al. “Influence of awareness of the recording of breathing on respiratory pattern in healthy humans”, *European Respiratory Journal*, v. 10, n. 1, pp. 161–166, jan. 1997.
- [3] ALNOWAMI, M., ALNWAIMI, B., TAHAVORI, F., et al. “A quantitative assessment of using the Kinect for Xbox 360 for respiratory surface motion tracking”, *Proc. SPIE 8316, Medical Imaging 2012: Image-Guided Procedures, Robotic Interventions, and Modeling, 83161T*, fev. 2012.
- [4] XIA, J., SIOCHI, R. A. “A real-time respiratory motion monitoring system using KINECT: Proof of concept”, *Medical physics*, v. 39, n. 5, pp. 2682–2685, 2012.
- [5] AOKI, H., MIYAZAKI, M., NAKAMURA, H. “Non-contact respiration measurement using structured light 3-D sensor”, *SICE Annual Conference (SICE), 2012 Proceedings of*, pp. 614–618, 2012.
- [6] LUSTOSA, L. A. *Monitorização não invasiva da frequência respiratória por método optoeletrônico: ensaios in vitro com o Kinect*. Dissertação de mestrado, Universidade Federal do Rio de Janeiro / COPPE, Rio de Janeiro, Brasil, 2013.
- [7] GUYTON, A. C., HALL, J. E. *Tratado de Fisiologia Médica*. 11 ed. Rio de Janeiro, Saunders - Elsevier, 2006.
- [8] MICROSOFT. “Kinect for Windows SDK 1.8”, 2015. Disponível em: <<https://msdn.microsoft.com/en-us/library/jj131033.aspx>>.
- [9] ANDERSEN, M. R., JENSEN, T., LISOUSKI, P., et al. *Kinect Depth Sensor Evaluation for Computer Vision Applications*. In: Report, Technical



report ECE-TR-6. Department of Engineering, Aarhus University, Denmark., 2012.

- [10] SHOTTON, J., FITZGIBBON, A., COOK, M., et al. *Real-Time Human Pose Recognition in Parts from Single Depth Images*. IEEE, June 2011. Disponível em: <<http://research.microsoft.com/apps/pubs/default.aspx?id=145347>>.
- [11] WEISSTEIN, E. W. “Plane”, *From MathWorld—A Wolfram Web Resource* <http://mathworld.wolfram.com/Plane.html> [Acesso em 01 05 2015], 2015.
- [12] WEISSTEIN, E. W. “Point-Plane Distance”, *From MathWorld—A Wolfram Web Resource* <http://mathworld.wolfram.com/Point-PlaneDistance.html> [Acesso em 01 05 2015], 2015.
- [13] MICROSOFT. “Introduction to WPF”, 2015. Disponível em: <<https://msdn.microsoft.com/en-us/library/aa970268%28v=vs.110%29.aspx>>.
- [14] HARIDI, S., FRANZÉN, N. “7. Modules and Interfaces’ Mozart Documentation”, <http://www.mozart-oz.org/documentation/index.html> [Acesso em 17 01 2015], 2015.
- [15] PINO, A. V., L. T. KAGAMI, L. T., JANDRE, F. C. “DAS - um Programa de Aquisição e Processamento de Sinais para Engenharia Pulmonar”, *Proceedings of the III Congresso Latino Americano de Engenharia Biomédica*, 2004.
- [16] PINO, A. V., L. T. KAGAMI, L. T., JANDRE, F. C. “Mecânica - um sistema para análise do sistema respiratório”, *Anais do XVIII Congresso Brasileiro de Engenharia Biomédica*, 2002.
- [17] ALTMAN, D., BLAND, J. “Measurement in medicine: the analysis of method comparison studies”, *The Statistician*, , n. 32, pp. 307–317, 1983.