

UNIVERSIDADE DO SAGRADO CORAÇÃO

EDMAR SALDANHA DE MATTOS

**DESENVOLVIMENTO DE UM DISPOSITIVO
TERMOGRÁFICO PARA VISUALIZAÇÃO DE
TEMPERATURAS UTILIZANDO ARDUINO**

BAURU
2015

EDMAR SALDANHA DE MATTOS

**DESENVOLVIMENTO DE UM DISPOSITIVO
TERMOGRÁFICO PARA VISUALIZAÇÃO DE
TEMPERATURAS UTILIZANDO ARDUINO**

Trabalho de Conclusão de Curso apresentado ao Centro de Ciências Exatas e Sociais Aplicadas como parte dos requisitos para obtenção do título de Bacharel em Ciência da Computação, sob orientação do Prof. Esp. Andre Luiz Ferraz Castro.

BAURU
2015

M4447d	<p data-bbox="518 1462 901 1500">Mattos, Edmar Saldanha de</p> <p data-bbox="518 1534 1300 1646">Desenvolvimento de um dispositivo termográfico para visualização de temperaturas utilizando Arduino / Edmar Saldanha de Mattos. -- 2015.</p> <p data-bbox="566 1646 678 1680">52f. : il.</p> <p data-bbox="566 1713 1236 1747">Orientador: Prof. Esp. André Luiz Ferraz Castro.</p> <p data-bbox="518 1780 1300 1892">Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Universidade do Sagrado Coração – Bauru – SP.</p> <p data-bbox="518 1926 1300 1993">1. Termografia. 2. Arduino. 3. Delphi. 4. Varredura. 5. Infravermelho. I. Castro, André Luiz Ferraz. II. Título.</p>
--------	---

EDMAR SALDANHA DE MATTOS

**DESENVOLVIMENTO DE UM DISPOSITIVO TERMOGRÁFICO PARA
VISUALIZAÇÃO DE TEMPERATURAS UTILIZANDO ARDUINO**

Trabalho de Conclusão de Curso apresentado ao Centro de Ciências Exatas e Sociais Aplicadas como parte dos requisitos para obtenção do título de Bacharel em Ciência da Computação, sob a orientação do Prof. Esp. Andre Luiz Ferraz Castro.

Banca examinadora:

Prof. Esp. Andre Luiz Ferraz Castro
Universidade Sagrado Coração

Prof. M.e Patrick Pedreira Silva
Universidade Sagrado Coração

Prof. Dr. Elvio Gilberto da Silva
Universidade Sagrado Coração

Bauru, 27 de novembro de 2015

RESUMO

Desde a revolução industrial, quando ocorreu a transição de métodos de produção artesanais para a produção por máquinas, o mundo tem se transformado em uma corrida pelo consumismo com as indústrias produzindo produtos em larga escala reduzindo seus custos e modificando a forma de trabalho de muitas pessoas. Esse consumismo foi crescendo e até os dias atuais funciona de maneira altamente competitiva, onde cada vez mais quem produz mais consegue melhores negócios e maiores lucros. Seguindo esse crescimento, também houve a necessidade de aumentar também o controle dos processos, de logística e não menos importante, a manutenção dos equipamentos de produção, pois com as máquinas paradas mesmo que por uma fração de tempo significa um grande prejuízo financeiro. Um dos equipamentos de manutenção mais modernos e menos invasivos utilizados na indústria nos últimos anos são os termógrafos. Eles mostram com clareza alguns defeitos e vazamentos em máquinas onde não é possível ver com outros equipamentos não invasivos. Este trabalho desenvolveu um dispositivo e software de visão termográfica por varredura de pontos de baixo custo utilizando poucos componentes eletrônicos que qualquer pessoa com algum conhecimento em eletrônica poderia reproduzi-lo com facilidade. O dispositivo funcionou de maneira adequada, o software também funcionou adequadamente, porém o sensor escolhido no trabalho não teve um desempenho aceitável para a aplicação.

Palavras-chave: Termografia. Arduino. Delphi. Varredura. Pontos. Infravermelho.

ABSTRACT

Since the industrial revolution, when the transition occurred artisanal production methods for the production by machines, the world has turned into a race by consumerism with industries producing large-scale products reducing costs and changing the way of work of many people . This consumption was growing and until today operates in a highly competitive manner, where more and more those who produce more can best business and higher profits. Following this growth, there was also the need to also increase process control, logistics and not least, the maintenance of production equipment, as with parades machines even for a fraction of time means a big financial loss. One of the most modern and less invasive maintenance of equipment used in the industry in recent years are thermal imagers. They clearly show some defects and leaks in machines where you can not do with other non-invasive equipment. In this work was developed a device and software thermal vision by point scanning cheap using few electronic components that anyone with some knowledge in electronics could do it easily. The device worked properly, the software also worked properly, but the chosen sensor in the study was not acceptable performance for the application.

Keywords: Termography. Arduino. Delphi. Scanning. Point. Infrared.

LISTA DE FIGURAS

Figura 1 - Espectro eletromagnético	14
Figura 2 - Imagem termográfica de equipamentos diversos	16
Figura 3 - Componentes de um sistema de processamento de imagens.....	17
Figura 4 - Arduino Uno R2 (Placa de desenvolvimento)	18
Figura 5 - Ambiente de desenvolvimento	22
Figura 6 - Diferença de componente visual e não visual.....	23
Figura 7 - Uma alusão do funcionamento do termógrafo a ser desenvolvido	28
Figura 8 - Varredura de pontos x e y.....	28
Figura 9 - Um exemplo de representação termográfica e sua escala de cores	29
Figura 10 - Diagrama de Caso de Uso.....	31
Figura 11 - Diagrama de Atividades	31
Figura 12 - Módulo do sensor de temperatura (MLX90614).....	32
Figura 13 - Parte de trás do módulo MLX90614.....	33
Figura 14 - Conector feito com barra de terminais fêmeas	33
Figura 15 - Ligação do módulo MLX90614 ao Arduino	34
Figura 16 - IDE do Arduino com o exemplo adicionado pela biblioteca	35
Figura 17 - Código gerado a partir do exemplo da biblioteca	36
Figura 18 - Comandos da biblioteca comentados	36
Figura 19 - Janela do "Monitor Serial" mostrando as temperaturas lidas	37
Figura 20 - Medidor de temperatura "à laser".....	38
Figura 21 - Lente do termômetro comercial	39
Figura 22 - Lente do termômetro comercial	39
Figura 23 - Servo motor utilizado neste trabalho.....	40
Figura 24 - Código teste do servo motor comentado	41
Figura 25 - Servo motores montados para fazer a varredura de x e y	42
Figura 26 - Código fonte completo e comentado da solução encontrada para o Arduino.....	43
Figura 27 - Software desenvolvido	44
Figura 28 - Software desenvolvido em funcionamento.....	45
Figura 29 - Gráfico comparativo de temperaturas frias	46
Figura 30 - Gráfico comparativo de temperaturas quentes	47

LISTA DE TABELAS

Tabela 1 – Especificações do Arduino UNO	20
Tabela 2 - Modelo mais barato e mais caro encontrados em uma breve pesquisa...	27
Tabela 3 - Representação dos dados recebidos pelo sensor.....	30
Tabela 4 - Especificações do medidor de referência.....	38
Tabela 5 - Ligação dos componentes no Arduino	41
Tabela 6 - Comparativo de temperaturas frias	46
Tabela 7 - Comparativo de temperaturas quentes	47

LISTA DE ABREVIATURAS E SIGLAS

DC - Direct Current

IDE - Integrated Development Environment

LCD - Liquid Crystal Display

PWM - Pulse-Width Modulation

RAD - Rapid Application Development

UML - Unified Modeling Language

USB - Universal Serial Bus

Wi-Fi - Wireless Fidelity

SUMÁRIO

1	INTRODUÇÃO	10
2	OBJETIVOS	12
2.1	OBJETIVO GERAL.....	12
2.2	OBJETIVOS ESPECÍFICOS	12
3	REVISÃO DA LITERATURA.....	13
3.1	TERMOGRAFIA	13
3.2	PROCESSAMENTO DE IMAGENS	17
3.3	ARDUINO.....	18
3.4	DELPHI	20
3.5	ESTRUTURA DO IDE	21
3.6	TRABALHOS CORRELATOS	23
4	MATERIAIS E MÉTODOS.....	25
4.1	MATERIAIS.....	25
4.1.1	Softwares utilizados	25
4.1.2	Hardwares utilizados.....	25
4.1.3	Funcionamento	26
4.1.4	ENGENHARIA DE SOFTWARE	30
4.2	DESENVOLVIMENTO.....	32
5	TESTES E RESULTADOS.....	43
6	CONSIDERAÇÕES FINAIS	48
	TRABALHOS FUTUROS	49
	REFERÊNCIAS.....	50

1 INTRODUÇÃO

Desde o início da revolução industrial, onde se passou a transição de métodos de produção artesanais, a indústria teve grandes mudanças tanto em suas capacidades, quanto em suas tecnologias. A produção em massa gerada pelas indústrias fizeram com que os produtos diminuíssem de valor e acarretaram o aumento do consumo, alavancado pelo crescimento dessas indústrias deu-se o surgimento da sociedade de consumo, e com o tempo fez com que cada vez mais se exigisse dos maquinários de produção em larga escala, que foram evoluindo com o tempo.

Desde as primeiras fábricas com suas máquinas mecânicas, sempre se teve problemas com a medição de determinadas variáveis para verificar o funcionamento ou prever um possível defeito que venha causar algum dano nos equipamentos e conseqüentemente atrapalhar a produção e até mesmo pará-la.

Conforme o tempo foi passando, a crescente e inevitável automação que se observa nos dias atuais, cada vez mais se preocupa com velocidade e confiabilidade de sistemas automáticos para aumentar o rendimento de produção e diminuição de riscos e prevenção de falhas.

As indústrias foram crescendo a tal ponto, que hoje em dia, uma máquina parada por uma fração de tempo pode causar um prejuízo financeiro consideravelmente grande, devido às técnicas empregadas para diminuir custos e aumentar a produção, e a detecção antecipada de possíveis falhas é evidente e fundamental para aumentar a flexibilidade e autonomia do sistema industrial.

Seguindo esse conceito, muitos sensores eletrônicos foram desenvolvidos na área industrial para aferir níveis de líquidos, temperaturas de máquinas, velocidade de motores, dentre outros.

A atividade de manutenção preditiva aumentam a disponibilidade do equipamento/máquina e das instalações, reduzindo o número de inesperadas degradações ou interrupções do serviço, analisando parâmetros tais como vibração, temperatura e fluxo e utilizando-se de alguma tecnologia, diferentemente da manutenção preventiva, onde o foco é em tarefas manuais. (WIREMAN, 2005).

O papel da manutenção mostra-se essencial na garantia da qualidade e da produtividade das empresas.

A manutenção preventiva e preditiva deve ser criteriosa, pois a falta de critério leva a custos excessivos, devido ao aumento de falhas ou intervenções em excesso

Manutenção preditiva caracteriza-se pela medição e análise de variáveis da máquina que possam prognosticar uma eventual falha (LIMA; MARCORIN, 2003), e fazer com que se encontre uma limiar na manutenção que possa atender melhor os padrões exigidos no mercado atual, sem deixar que falhas inesperadas parem o processo de produção.

Com o advento da eletrônica, os sensores eletrônicos, e novas maneiras de inspeção de máquinas vieram e com isso foi diminuindo o uso de manutenção corretiva, e a manutenção preditiva tornou-se o ponto chave estratégico em diversos tipos de negócio.

A medição da temperatura é uma técnica de manutenção que foi utilizada desde o início do desenvolvimento industrial, pois identifica a quantidade de calor presente nos equipamentos e seus possíveis defeitos.

A termografia é uma das novas técnicas que está fortemente sendo utilizada no mercado, não só na indústria, como em outras grandes áreas de conhecimento como: engenharias (civil, elétrica, mecânica, mecatrônica, computação, etc.), medicina veterinária, agronomia, medicina, dentre outros.

A técnica de termografia foi utilizada neste trabalho para o desenvolvimento de um dispositivo de baixo custo, utilizando a plataforma Arduino e um software capaz de criar uma visão termográfica de qualquer objeto que possa emitir radiação infravermelha. Podendo ser utilizado tanto na indústria para aferir temperaturas, em diagnósticos médicos, engenharia civil, dentre outras áreas diversas.

Com a visão termográfica é possível facilmente perceber onde ocorrem as variações térmicas do objeto a ser analisado para poder tomar decisões mais acertadas sobre seu funcionamento.

2 OBJETIVOS

Desenvolver um protótipo de dispositivo e software para geração de imagem termográfica para visualização da temperatura de objetos utilizando poucos componentes de baixo custo e software simples e intuitivo.

2.1 OBJETIVO GERAL

Desenvolver um dispositivo termográfico para visualização de temperaturas utilizando Arduino.

2.2 OBJETIVOS ESPECÍFICOS

- a) Estudar métodos de criação de imagens termográficas;
- b) Levantar dados sobre o funcionamento de dispositivos de termografia;
- c) Pesquisar formas de obtenção de dados para visão termográfica;
- d) Pesquisar técnicas de criação e processamento de imagens a partir de dados obtidos;
- e) Obter informações sobre cores e temperaturas na termografia;
- f) Produzir um mecanismo para obtenção de dados térmicos;
- g) Desenvolver software de controle e obtenção de dados térmicos;
- h) Testar e analisar os dados obtidos.

3 REVISÃO DA LITERATURA

3.1 TERMOGRAFIA

A radiação eletromagnética é definida como a emissão ou transferência de energia em forma de ondas eletromagnéticas ou partículas. (AZEVEDO; CONCI; LETA, 2008).

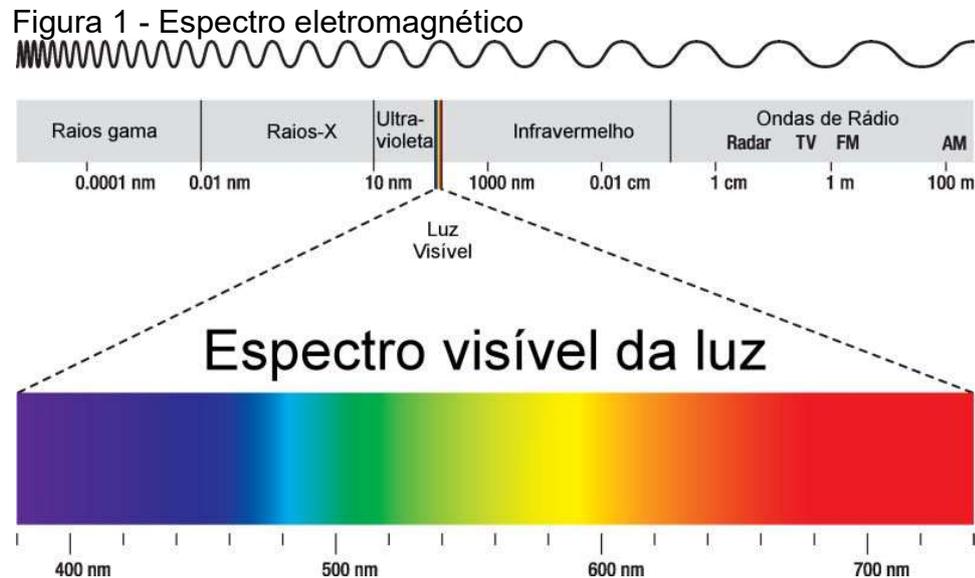
A descoberta da radiação infravermelha é atribuída a Willian Herschel, um Astrônomo, antes do século 19. Herschel tentava descobrir quais as cores do espectro que eram responsáveis pelo aquecimento de objetos, usando um prisma para refletir a luz do sol, conclui que a temperatura aumentava a medida que a luz passava da cor violeta para a vermelha, e a maior temperatura ocorria na faixa do além do visível, sobre o qual Herschel nomeou de raios caloríficos, são hoje conhecidos como raios infravermelhos. (MENDONÇA, 2005).

O comprimento de onda do infravermelho no espectro eletromagnético situa-se faixa que vai de 2 a 12 μ m, de acordo com a Figura 1, sendo impossível de ser detectado pelo olho humano por ser um comprimento de onda muito longo, apenas se percebe seus efeitos sob a forma de calor ou frio. Ao contrário do comprimento de luz visível visto na Figura 1, qualquer corpo acima de zero grau absoluto 0°K (-273°C) emite radiação infravermelha, sendo assim cubos de gelo emitem infravermelho. Quanto mais alta a temperatura de um objeto maior a radiação infravermelha emitida. Por isso quanto maior a radiação infravermelha emitida, maior será a temperatura medida. (INFRATHERM, 2010).

O olho atua como um seletor sensível a uma faixa de comprimento de onda do espectro eletromagnético, tal como um sensor eletrônico, o olho está configurado para captar determinada frequência de ondas eletromagnéticas, e acima ou abaixo desse comprimento de onda o olho não é capaz de captar estas energias. O espectro de luz que o olho humano é capaz de captar pode ser visto na Figura 1, e a radiação Infravermelha, que foi analisada neste trabalho está acima da faixa de onda visível ao olho humano, porém pode ser percebida, em forma de calor. (AZEVEDO; CONCI; LETA, 2008).

Qualquer material pode absorver e emitir radiação infravermelha com base nos fenômenos de agitação molecular e a radiação térmica (infravermelha) é a radiação eletromagnética emitida por todos os objetos, e a sua origem reside, em última

instância no movimento aleatório dos átomos que os constituem. As moléculas de qualquer objeto a temperatura acima do zero absoluto (-273°C) radiam no infravermelho ainda que com intensidade reduzida (HECHT, 2002, p. 104).



Fonte: Carvalho (2006).

Termografia é um ensaio não destrutivo, que tem como princípio a medição de temperaturas pelo método de radiação infravermelha. Ela é uma técnica que passou nos últimos anos a ser muito utilizada, pois muitas falhas em equipamentos são antecidas pela geração e troca de calor, fazendo com que seja possível a detecção antecipada destas falhas.

Com a termografia se focalizam os problemas que devem ser corrigidos utilizando as técnicas convencionais e podendo encontrar outros problemas que em circunstâncias normais não seriam detectados. Dado que a termografia infravermelha é um meio que permite identificar, sem contato algum, componentes elétricos e mecânicos mais quentes do que deveriam estar, constituindo provavelmente uma área de falha e indicando também perdas excessivas de calor, provavelmente uma falha de um defeito isolado. (MOURA; MOURA, 2014, p. 15).

O trabalho de Santos e Pereira (2010) cita algumas aplicações para termografia em manutenção preditiva para monitoramento de máquinas evitando problemas como a fadiga de materiais com a visão termográfica aplicada em componentes principalmente como ferramenta de avaliação não destrutiva.

Os autores também ressaltam que:

Na termografia o perfil térmico do equipamento é mostrado em um gráfico de cores, sendo que as mudanças térmicas são observadas no termograma a partir das alterações na intensidade das cores. Essas alterações podem apontar problemas como mau funcionamento ou mesmo sinais de redução da vida útil do equipamento.

As ferramentas de termografia têm sido largamente utilizadas para monitorar as assinaturas térmicas de máquinas e equipamentos, pela sua praticidade não destrutiva e não necessitar o desligamento dos equipamentos em manutenção. (PEREIRA; SANTOS, 2010).

E segundo Mendonça (2005), a imagem infravermelha deve ser acompanhada por uma medição precisa para poder refletir as condições reais de um objeto.

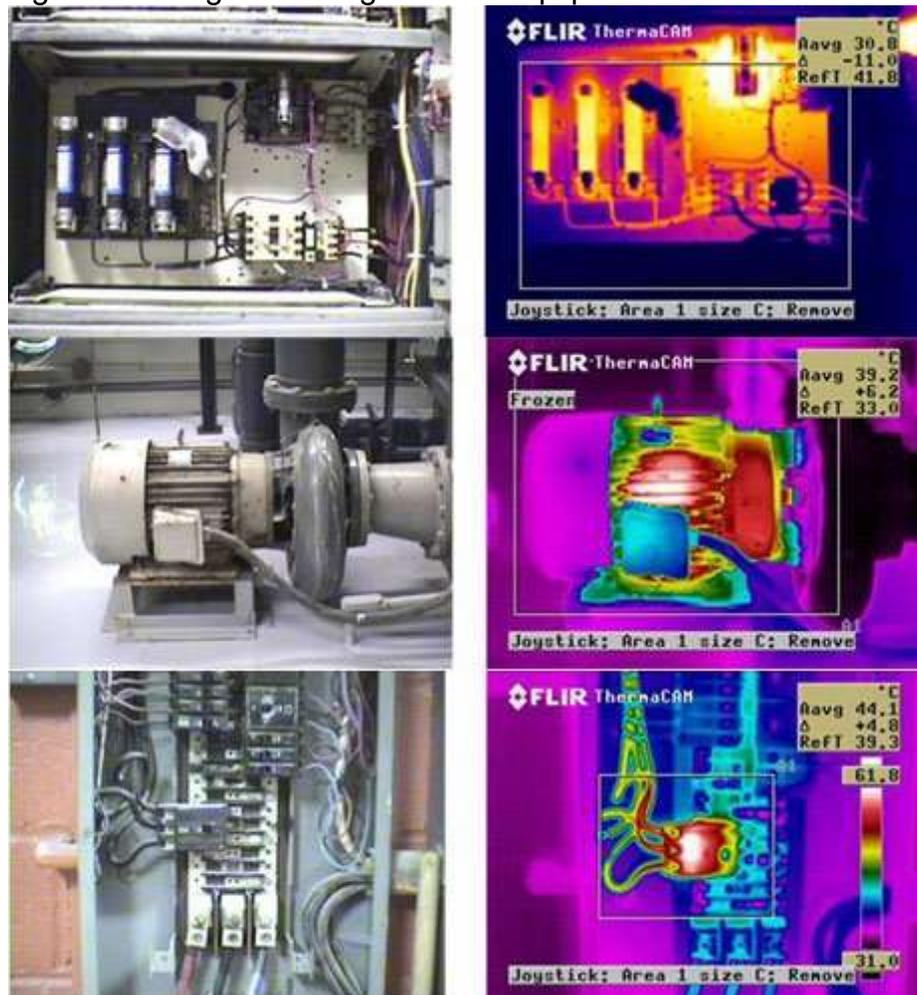
O aparelho de termografia detecta essa energia infravermelha (radiação de calor) por meio de um sensor, e a “converte em sinal elétrico”, depois com as medições de temperatura produz a imagem termográfica. (MENDONÇA, 2005). Um aparelho de termografia profissional gera imagens termográficas dos equipamentos a serem analisados, que pode ser visualizada na Figura 2.

A imagem gerada na Figura 2 pode mostrar claramente por meio de uma paleta de cores as mínimas e máximas da temperatura dos objetos captados pelo termógrafo profissional, pode-se ver o calor que radia dos objetos, cada temperatura com sua cor correspondente, podendo detectar alguma falha e ajudando a encontrar algum defeito que dificilmente poderia ser detectado por outros meios senão utilizando-se da termografia.

Esse é basicamente o funcionamento de um termógrafo, além da geração da imagem explícita com suas cores correspondentes às temperaturas, é possível verificar em cada ponto da imagem sua temperatura específica.

O termógrafo capta basicamente a superfície do qual está sendo avaliada, e apesar de não “ver” internamente o objeto lido, de acordo com a segunda lei da termodinâmica, “é proibida a transferência espontânea de calor dos corpos mais frios para os mais quentes. A transferência no sentido oposto é o sentido natural e se processa até que todos os corpos atinjam o mesmo estado térmico”. (GASPAR, 2002).

Figura 2 - Imagem termográfica de equipamentos diversos



Fonte: Chiodini (c2015).

Então, de acordo com essa lei, um exemplo: caso uma máquina esteja gerando calor em seu interior mediante um defeito, por força da termodinâmica, também estará transferindo calor por meio de condução para sua superfície, irradiando ondas infravermelhas, não imediatamente, mas depois de ligado por um tempo em seu funcionamento pleno.

Também é preferível, caso possível, que o objeto a ser medido com um termógrafo tenha ficado por um tempo em um ambiente com temperatura controlada, para que ocorra o equilíbrio térmico do objeto com o ambiente, onde foi feita a imagem termográfica de pessoas durante o exercício, elas foram adaptadas à temperatura da sala durante 30 minutos. (FERNANDES, 2013).

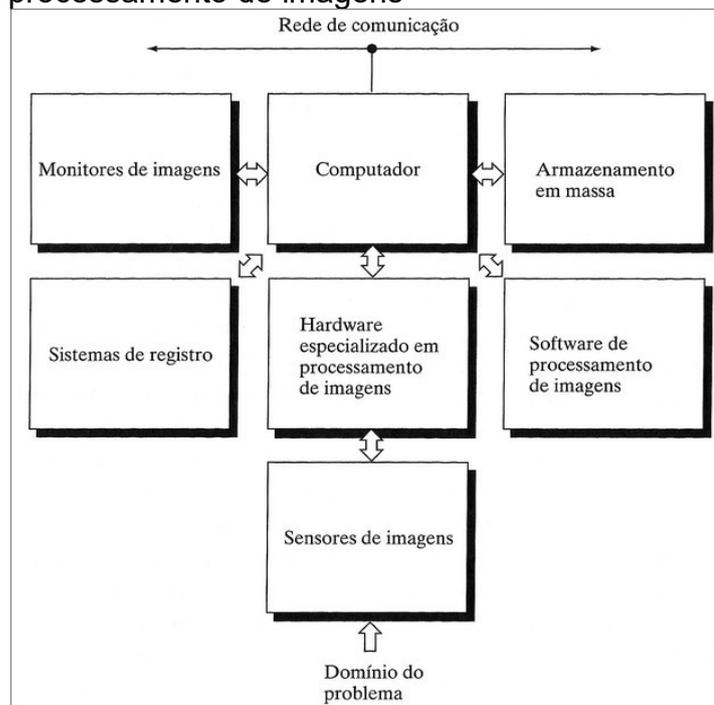
A termografia é utilizada em várias áreas, como manutenção em quadros elétricos, inspeção de fornos e caldeiras em indústrias, análise de temperatura em placas de circuitos, na construção para detectar vazamentos de ar e isolamentos em

prédios, em mecânica para avaliar quais partes de um motor estão com aumento anormal de temperatura, na indústria automobilística e ultimamente vem sendo largamente utilizada para fins de diagnósticos médicos.

3.2 PROCESSAMENTO DE IMAGENS

Processamento de imagem, segundo Gonzalez e Woods, se refere ao processamento de imagens digitais por um computador digital. Também, os autores definem que “Uma imagem pode ser definida como uma função bidimensional, $f(x,y)$, em que x e y são coordenadas espaciais (plano)”. (GONZALES e WOODS, 2010).

Figura 3 - Componentes de um sistema de processamento de imagens



Fonte: (GONZALES e WOODS, 2010)

Na Figura 3 podemos ver os componentes básicos que constituem um sistema de uso geral típico para o processamento de imagem cada qual com sua função, porém as funções realmente necessárias a este trabalho serão listadas a seguir:

- O Sensoriamento é o elemento que adquire a imagem, um dispositivo físico que é sensível à energia irradiada pelo objeto a ser capturado.

- O Hardware é o digitalizador, o componente que converte a saída do dispositivo sensor em dados digitais;
- O Computador recebe os dados, armazena em dispositivo de armazenamento em massa para o processamento;
- O Software que realiza o processamento da imagem em si, converte os dados recebidos armazenados e gera imagem para o monitor de imagens.

Os componentes do sistema indicados por Gonzalez e Woods, podem ser facilmente associados ao trabalho proposto, após a leitura dos capítulos seguintes, onde o sistema de sensoriamento foi o sensor MLX90614, o Arduino como hardware digitalizador e o software que foi desenvolvido, utilizando um computador com armazenamento em massa para receber os dados e gerar a imagem. Todos esses componentes formam um sistema de processamento de imagem completo, tal como foi elucidado pelos autores.

3.3 ARDUINO

Arduino é uma plataforma de prototipagem de código aberto baseado em hardware fácil de utilizar e software, é uma plataforma de computação física open-source baseada em uma placa de microcontrolador simples, e um ambiente de desenvolvimento para escrever software para a placa. (ARDUINO, c2015, tradução nossa).

Figura 4 - Arduino Uno R2 (Placa de desenvolvimento)



Fonte: Arduinoboarduno (c2015).

Com o Arduino (Figura 4) pode-se criar vários projetos interativos, utilizando entradas a partir de uma variedade de sensores ou interruptores, e controlar uma variedade de luzes, motores, telas de LCD entre outras saídas físicas.

Os projetos feitos com o Arduino podem se comunicar com o software em execução no computador por comunicação serial via porta USB, via Bluetooth, WiFi e outros, em tempo real, pode também funcionar sem o uso do computador, autonomamente utilizando baterias ou ligado em uma fonte DC.

Seu projeto é de código aberto, baseado normalmente no microcontrolador ATmega, produzido pela empresa Atmel Corporation, pode ser montado em casa, comprando-se os componentes eletrônicos (microcontrolador, resistores, capacitores, etc.) e modificado e comercializado de qualquer maneira, e também pode ser comprado em lojas especializadas no mundo todo, além de possuir IDE própria para desenvolvimento.

Originalmente, foi concebido para ser utilizado como ferramenta para projetos de computação por designers e estudantes de arte [...] tem sido adotado como a ferramenta preferida das comunidades de desenvolvedores e fabricantes interessados na construção e prototipagem de seus próprios projetos. (EVANS; HOCHENBAUM; NOBLE, 2014, p.24).

Utiliza a linguagem de programação baseada em Wiring, semelhante à linguagem C/C++, não sendo necessário conhecimento profundo em linguagens de programação de baixo nível ou mesmo grandes conhecimentos em programação, pois a própria IDE do Arduino contém exemplos de como utilizar a maioria dos sensores e componentes que o Arduino é capaz de suportar.

Existem vários tipos de placas de desenvolvimento do Arduino (Arduino UNO, Mega, Leonardo, LilyPad, dentre outros), e para este trabalho foi escolhido o Arduino UNO (Figura 4), pois tem quantidade de portas mais que necessárias para o desenvolvimento do projeto, é a placa mais fácil de ser encontrada no mercado por ser a mais conhecida e difundida. Seguem as especificações de um Arduino UNO na Tabela 1.

Tabela 1 – Especificações do Arduino UNO

Microcontrolador	ATmega328
Tensão de operação (recomendada)	7-12V
Tensão de alimentação	6-20V
Pinos entrada e saída digitais	14 (6 podem ser usados como PWM)
Pinos entrada e saída analógicas	6
Corrente por pino I/O	40mA
Memória Flash	32 KB (ATmega328) onde 0.5 KB é usado pelo Bootloader
SRAM	2KB
EEPROM	1KB
Clock velocidade	16MHz
Interrupções externas	2
Comunicação serial	UART TTL (5V)

Fonte: (ARDUINO..., c2015)

Por ser uma plataforma de desenvolvimento de baixo custo e rápido desenvolvimento, o Arduino é também utilizado em vários projetos de trabalhos acadêmicos, e foi utilizado neste trabalho.

3.4 DELPHI

O Delphi é um compilador e uma IDE, produzido antigamente pela Borland Software Corporation e atualmente produzido pela Embarcadero. O Delphi, originalmente direcionado para a plataforma Windows, chegou a ser usado para desenvolvimento de aplicações nativas para Linux através do Kylix (o Kylix é um IDE para as linguagens C++ e Object Pascal), e para o framework Microsoft .NET em suas versões mais recentes. (EMBARCADERO..., c2015).

O ambiente de desenvolvimento do Delphi é baseado em uma extensão orientada a objetos da linguagem de programação Pascal antes conhecida como Object Pascal. Recentemente, a Embarcadero define esta linguagem como “a linguagem Delphi”. A maioria das linguagens de programação modernas suporta a programação orientada a objetos, ou POO (object-oriented programming – OOP), que são baseadas em três conceitos fundamentais: encapsulamento (normalmente

implementado com classes), herança e polimorfismo (ou ligação tardia). (CANTÙ, 2006).

O Delphi foi utilizado neste trabalho por ter um desenvolvimento rápido para computadores desktop, ter grande suporte à novos componentes desenvolvidos tanto por usuários como por comunidades ligadas ao desenvolvimento em Delphi. É inclusive o componente TComPort, que tratará de realizar a comunicação com a parte de hardware e foi o principal intermediador para comunicação com o software desenvolvido.

A estrutura e o funcionamento do IDE do Delphi serão apresentados no tópico a seguir.

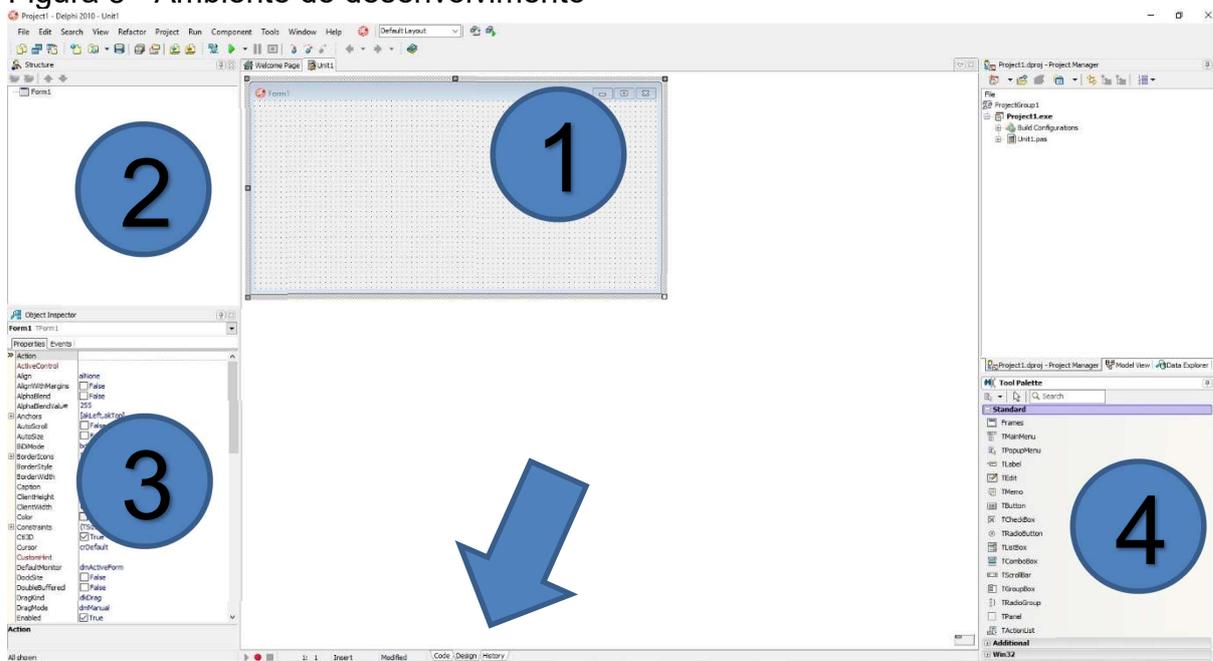
3.5 ESTRUTURA DO IDE

O ambiente de desenvolvimento do Delphi é de fácil assimilação, quando um novo projeto é criado, é evidenciado na área central as abas “Design”, onde serão colocados os botões, campos editáveis e outros componentes visuais e também não visuais, como o que foi utilizado para comunicação serial (TComport) no programa a ser desenvolvido. Realmente é a parte de design do programa, como será o seu layout que será apresentado para o usuário do programa desenvolvido, evidenciado com o número 1 na Figura 5, onde se pode mover os componentes inseridos, alterando seu tamanho e posição. Outra aba importante que fica na região central é a aba “Code”, onde fica todo o código fonte do programa desenvolvido, funções e procedimentos, que pode ser acessada clicando na sua respectiva aba (evidenciado na Figura 5 com uma seta) ou mesmo ao dar um duplo clique em cima de um componente para acessar sua função específica de clique.

Também na Figura 5, a área esquerda acima evidenciado com o número 2, há a janela “Structure”, onde ficam todos os componentes visuais e não visuais inseridos de forma hierárquica, facilitando a seleção e até mesmo uma alteração de hierarquia onde os componentes estão inseridos, também facilita na hora de selecionar um componente que está sob o outro. Já ligeiramente abaixo (círculo número 3, na Figura 5) está a janela “Object Inspector”, onde ao selecionar um componente, pode-se alterar suas propriedades (tamanho, nome do componente, texto, etc.), e por fim, no círculo 4 estão os componentes que foram inseridos no programa, sejam eles visuais ou não.

O Delphi possui outras abas e janelas, porém a explicação mostrada é o básico do ambiente de desenvolvimento, não entrando em detalhes outras funções e janelas mais necessárias para um estudo mais aprofundado.

Figura 5 - Ambiente de desenvolvimento



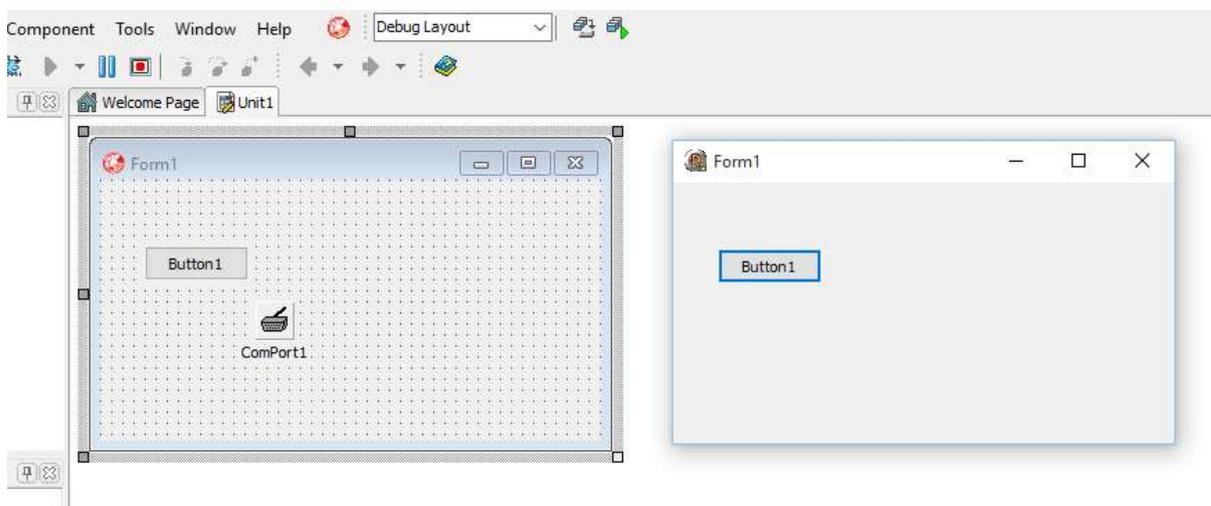
Fonte: Elaborada pelo autor.

O componente que foi utilizado para comunicação serial com o Arduino é um componente não visual, chamado ComPort, Figura 6, onde do lado esquerdo, está o desenvolvimento do programa no Delphi, e do lado direito o programa em execução, onde pode se notar o que componente ComPort1 não aparece na janela.

O ComPort é um componente não nativo do Delphi, é de código fonte aberto sob licença Public Domain, pode ser baixado livremente no site SourceForge.net e pode ser instalado em várias versões do Delphi. A versão que foi utilizada neste trabalho é a versão 4.11, com última atualização em 30/04/2013. (CRNILA, 2003).

Neste trabalho foi usados vários componentes, como o TButton, TImage, TListBox, dentre outros para compor o software a desenvolver, porém, foi usado o mínimo de componentes externos possível para facilitar o desenvolvimento e torna-lo o mais “universal” possível.

Figura 6 - Diferença de componente visual e não visual



Fonte: Elaborada pelo autor.

3.6 TRABALHOS CORRELATOS

O trabalho proposto está dentre duas áreas de conhecimento de vasta utilização no momento: Arduino e Termografia, porém não foram encontrados estes dois elementos juntos em um único trabalho.

Muito se vê em trabalhos acadêmicos a utilização do Arduino para automação residencial ou mesmo a criação de robôs para o ensino da própria plataforma, o que não tem muita relação com o trabalho proposto.

O Arduino por ser uma plataforma de desenvolvimento muito barata, é utilizado em milhares de projetos no mundo todo, e como é de fácil utilização, geralmente é utilizado para projetos próprios (caseiros) ou comerciais, e não é tão utilizado no meio acadêmico tanto quanto poderia, e sempre é subutilizado em suas funções.

Quanto à termografia, esta técnica é largamente utilizada nos meios acadêmicos, pois possui muitas aplicações em várias áreas de conhecimento (mecânica, elétrica, eletrônica, saúde, zootecnia, informática, construção civil, dentre outras) e não tem basicamente nenhuma aplicação para um usuário comum, pois a termografia é geralmente utilizada na indústria e para fins comerciais.

Na grande parte dos materiais encontrados relativos à termografia, ela é utilizada na área da saúde para captar a temperatura do corpo e prever se há alguma

alteração em determinado local, porém sempre utilizando dispositivos comerciais para a obtenção das imagens, e não criando-as com dispositivos próprios.

Com isso, este trabalho vem contribuir com o acréscimo de informações acadêmicas nestas duas grandes áreas tão utilizadas, o Arduino e a termografia, para que se possa não ter só a junção dessas duas áreas, mas também com resultados concisos e análises dos resultados mais relevantes, e talvez contribuir para trabalhos futuros utilizando destes conhecimentos gerados por este trabalho.

4 MATERIAIS E MÉTODOS

O trabalho proposto visou utilizar de pesquisas exploratórias a fim de aprofundar os conhecimentos e proporcionar maior familiaridade das áreas de física: para analisar o funcionamento da radiação infravermelha; robótica: para criar o protótipo do termógrafo e programação em linguagem Pascal (Delphi) para a criação do programa que controlará o termógrafo e com isso produzindo resultados científicos de maneira satisfatória com o problema proposto inicialmente. Com isso, essas pesquisas foram baseadas nos três pilares deste trabalho: Termografia, Arduino e Delphi. Onde cada um tem suas particularidades para o estudo: a Termografia é uma ciência baseada em propriedades de ondas eletromagnéticas emitidas por todos os corpos, o Arduino é baseado em hardware/robótica e o Delphi é uma RAD que utiliza a linguagem Pascal para desenvolvimento de programas para Windows. (GIL, 2002).

4.1 MATERIAIS

4.1.1 Softwares utilizados

Para este trabalho foi utilizado a IDE (Integrated Development Environment) do Arduino para o desenvolvimento do programa (sketch) que fará o upload do arquivo para o chip do Arduino, disponível gratuitamente no site do Arduino, também foi utilizado a RAD (Rapid Application Development) do Delphi 2010, para o desenvolvimento do programa que exibirá a imagem termográfica gerada a partir dos dados do Arduino.

4.1.2 Hardwares utilizados

Para o desenvolvimento do trabalho foi utilizado:

- Um computador Intel® Core™ i5 750, 2,67GHz, 8GB de memória RAM, Sistema Operacional Windows 7 64bits, para o desenvolvimento do programa que controlará o termógrafo.
- Uma placa de desenvolvimento Arduino UNO
- Dois servo motores

- Um sensor de temperatura infravermelho modelo MLX96014
- Uma protoboard (placa de ensaios)
- Suporte para os motores e para o sensor
- Fios (jumpers).

4.1.3 Funcionamento

O sensor infravermelho foi escolhido para fazer as medições de temperatura, pois tem a característica de fazer medições sem contato com o objeto, fazendo com que se possa gerar a imagem térmica, e o modelo encontrado com essas características foi o MLX96014 produzido pela empresa Melexis Semiconductors, que tem as características procuradas para este trabalho.

Alguns pontos fortes deste sensor:

- Fácil integração, possui biblioteca pronta para uso (disponibilizado por terceiros);
- Tamanho pequeno, leve e de baixo custo;
- Faixa de medição de temperatura ampla: -40 a 125°C para temperatura ambiente e -70 a 380°C para o objeto medido;
- Precisão médica, resolução de medição de 0,01°C;
- Interface digital compatível para leituras de temperatura rápidas.

Também foi levado em conta o valor deste sensor, pois se não tivesse baixo custo, não seria viável sua aplicação nesta pesquisa, pois um termógrafo profissional tem faixa de valor que parte de aproximadamente USD900,00 até passar de USD29.000,00 dependendo de sua configuração. (FLIR, c2015)..

No mercado há poucos dispositivos de visão termográfica, e entre eles a empresa FLIR é referência no mercado de termógrafos profissionais, na Tabela 2 constam dois modelos, o mais barato e o mais caro encontrados no site da empresa.

As câmeras pesquisadas têm imagens termográficas dinâmicas em tempo real, e não estáticas como proposto neste trabalho.

Tabela 2 - Modelo mais barato e mais caro encontrados em uma breve pesquisa

Imagem	Modelo	Resolução	Valor (USD)
	FLIR E4	80x60	\$995.00
	FLIR T660	640x480	\$28,950.00

Fonte: Flir (c2015).

Com essas informações, partiu-se também da premissa de fazer um equipamento e software de baixo custo, onde também qualquer um com um pouco de conhecimento na área pode desenvolver um equipamento termográfico que gere imagem capaz de ficar o mais próximo possível de um equipamento profissional e que possa atender suas necessidades.

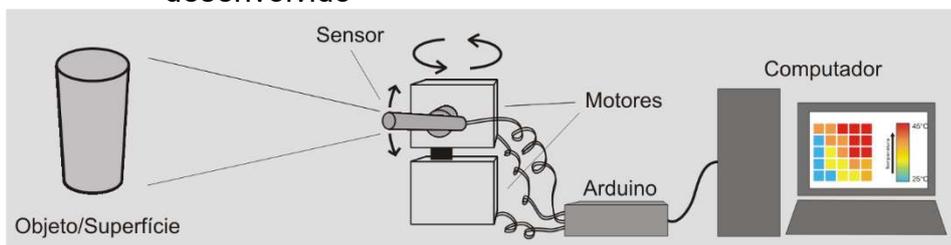
Foi preciso gerar uma imagem com um sensor de temperatura apenas e para que isso seja possível, foram adotados algumas soluções para o desenvolvimento do dispositivo deste trabalho.

“Para gerar uma imagem bidimensional (2-D) utilizando um único sensor, deve haver deslocamentos relativos, tanto na direção “x” quanto na direção “y” entre o sensor e a área de aquisição da imagem”. (GONZALES; WOODS, 2010, p. 30).

Utilizando-se dessa premissa, foi desenvolvido um suporte onde um servo motor controlará o eixo x (horizontal) e o outro controlará o eixo y (vertical) do sensor infravermelho de temperatura, a fim de capturar todos os pontos da área de interesse representado na Figura 7. Com esses pontos, que foram definidos como ponto (x, y,

temperatura) e teremos a coordenada x e y e sua temperatura lida pelo sensor infravermelho naquele ponto correspondente.

Figura 7 - Uma alusão do funcionamento do termógrafo a ser desenvolvido



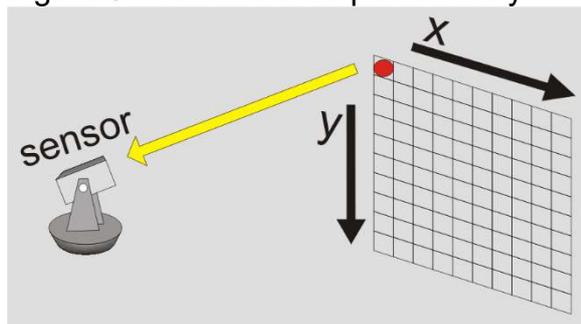
Fonte: Elaborada pelo autor.

Ao receber esses dados, o conjunto dessas localizações discretas serão a amostragem, que é determinado pelo arranjo do sensor utilizado para gerar a imagem, e quando é gerado por um único elemento sensor combinado com um movimento mecânico. Essa amostragem é realizada selecionando-se o número de incrementos mecânicos individuais em que o sensor será ativado para a coleta de dados (GONZALES e WOODS, 2010).

Ainda segundo Gonzales e Woods (2009, p. 35), “O Movimento mecânico pode ser muito exato, de forma que, em princípio, praticamente não existe um limite de precisão com a qual podemos amostrar uma imagem com esse procedimento”.

Os movimentos mecânicos realizados com os motores serão como uma varredura em uma tabela, onde fixado o valor de x (linha), varrerá todos os pontos de espaço determinado posteriormente pelo programa controlador, e passando por cada ponto (coluna) até o fim do objeto/superfície a ser medida, e logo em seguida, fará a movimentação nos motores a fim de direcionar o sensor à próxima linha, e novamente fará a varredura, e assim sucessivamente como mostra a Figura 8.

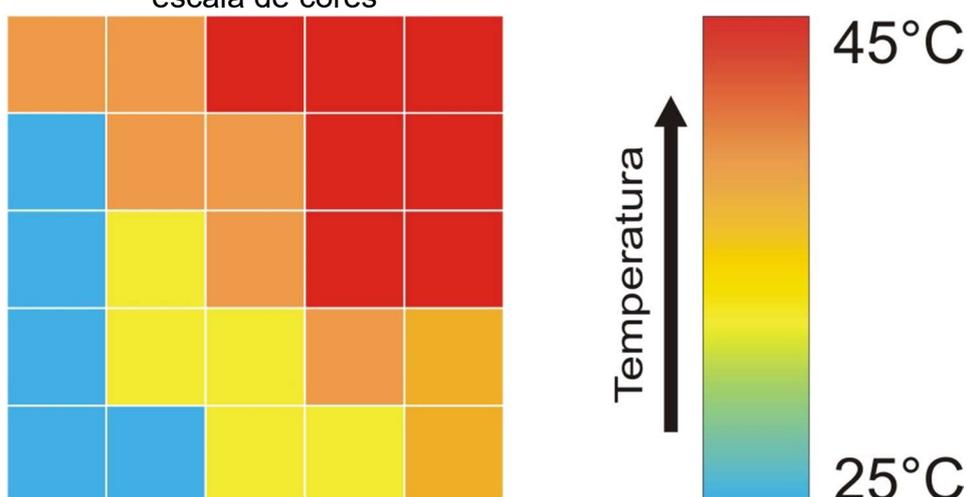
Figura 8 - Varredura de pontos x e y



Fonte: Elaborada pelo autor.

O programa desenvolvido em Delphi, controla o início e fim da captura dos pontos, e recebe os dados gerados dos pontos via comunicação serial gerada com o Arduino, e ligado no computador via cabo USB. O programa também faz uma imagem baseando-se nos pontos gerados, e irá gerar uma imagem baseando-se em uma escala de cores de acordo com a temperatura de cada ponto da imagem, onde a temperatura mais fria gerará uma imagem mais azul, uma temperatura mais quente, uma imagem mais vermelha, por exemplo. Depois de gerada as imagens, o programa também pode revelar as temperaturas correspondentes a cada ponto na imagem, ao clique do mouse à temperatura correspondente na imagem termográfica poderá ser visualizada.

Figura 9 - Um exemplo de representação termográfica e sua escala de cores



A Figura 9 representa as cores de uma imagem termográfica, onde cada cor representará uma faixa de temperatura diferente de acordo com os dados recebidos pelo sensor infravermelho, onde na Tabela 3 está a representação dos dados recebidos relativos à Figura 9.

De acordo com a Tabela 3, percebe-se que, como já citado anteriormente, cada ponto (x, y) lido conterá também o seu respectivo valor medido com o sensor infravermelho MLX90614, e formará os dados (linha, coluna, temperatura) e recebido os valores via USB no computador, ele foi convertido em uma imagem colorida de acordo com sua temperatura utilizando uma escala de cores, como mostra à direita da Figura 9.

Tabela 3 - Representação dos dados recebidos pelo sensor

	1ª Coluna	2ª Coluna	3ª Coluna	4ª Coluna	5ª Coluna
1ª Linha	(1,1,32)	(1,2,32)	(1,3,45)	(1,4,45)	(1,5,45)
2ª Linha	(2,1,25)	(2,2,32)	(2,3,32)	(2,4,45)	(2,5,45)
3ª Linha	(3,1,25)	(3,2,28)	(3,3,32)	(3,4,45)	(3,5,45)
4ª Linha	(4,1,25)	(4,2,28)	(4,3,28)	(4,4,32)	(4,5,32)
5ª Linha	(5,1,25)	(5,2,25)	(5,3,28)	(5,4,28)	(5,5,32)

Fonte: Elaborada pelo autor.

A imagem termográfica gerada é facilmente autoexplicativa, pois pode-se ver claramente as cores referentes às temperaturas, as imagens geradas no programa do computador são muito fáceis de entender, até mesmo por uma pessoa leiga no assunto.

4.1.4 ENGENHARIA DE SOFTWARE

Engenharia de software é uma ciência que estuda metodologias e padrões de desenvolvimento de software, e define métodos sistemáticos para o desenvolvimento de software, e busca melhorar e amadurecer as técnicas e ferramentas utilizadas no ambiente de desenvolvimento. (LOBO, 2008).

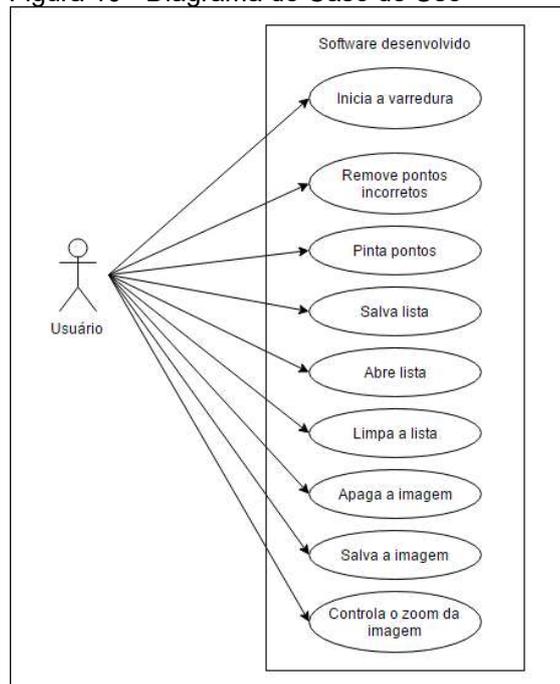
Métodos de desenvolvimento de software são propostos desde 1970, e possibilitam desenvolver sistemas em várias áreas de aplicação. A abordagem Orientada a Objetos tem uma grande aceitação devido à notação UML, e se tornou um padrão da indústria para do desenvolvimento de software. (HIMARA, 2012).

Para o desenvolvimento do software deste trabalho, foi analisado primeiramente as características básicas do funcionamento que seriam importantes para que o software pudesse realizar todas as funcionalidades esperadas no início do trabalho, e para isso foi desenvolvido uma modelagem UML do software.

A UML é uma linguagem para documentar, estruturar, ou mesmo estruturar visualmente as funções do software em diagramas padronizados.

Foi modelado um Diagrama de Caso de Uso (Figura 10) e um Diagrama de Atividades (Figura 11) com as funções desempenhadas pelo software.

Figura 10 - Diagrama de Caso de Uso

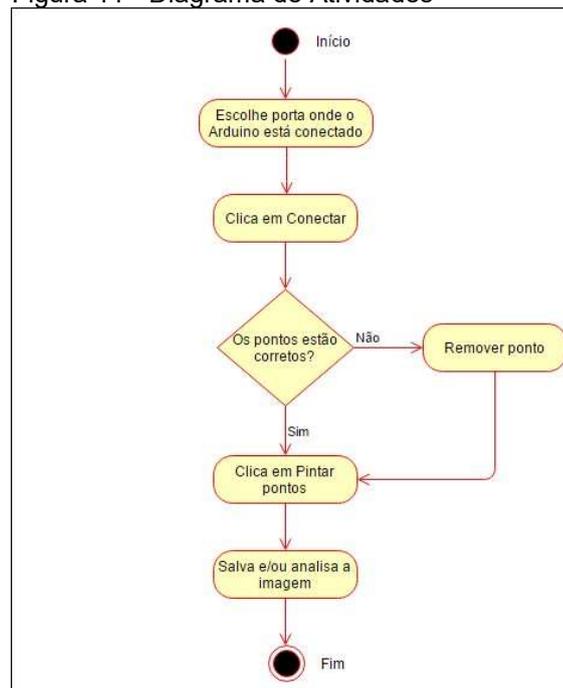


Fonte: Elaborada pelo autor.

O Diagrama de Caso de Uso descreve funcionalidades do sistema que foi projetado, e descreve a interação do Usuário (humano) com as funções do sistema.

Já o Diagrama de Atividades é um gráfico que descreve o fluxo de uma atividade para outra, para demonstrar dinamicamente os aspectos do sistema.

Figura 11 - Diagrama de Atividades



Fonte: Elaborada pelo autor.

Com os diagramas é possível ver facilmente como o software foi desenvolvido, com uma linguagem de fácil entendimento, de acordo com os diagramas que mostram as funcionalidades descritas.

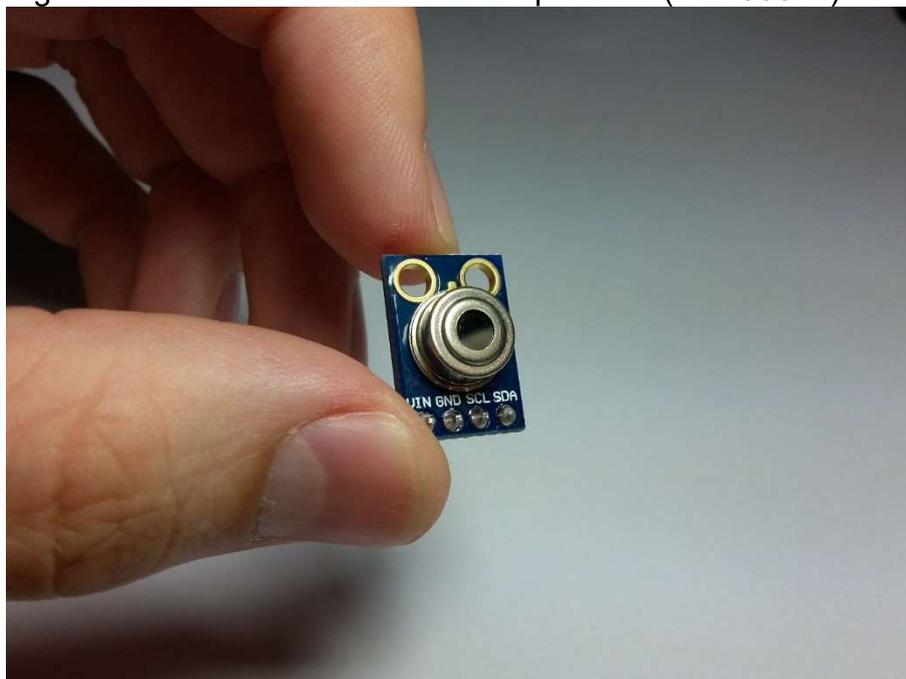
4.2 DESENVOLVIMENTO

Neste capítulo estão descritos os materiais e os métodos utilizados para o desenvolvimento do equipamento e do software utilizados nesse trabalho.

Os equipamentos em sua maioria já era de posse do autor e alguns itens apenas foram adquiridos ao longo do desenvolvimento do trabalho. As características dos equipamentos utilizados serão descritos ao longo do capítulo e também sua utilização para a obtenção dos resultados.

O principal componente utilizado para o desenvolvimento deste trabalho foi o sensor de temperatura infravermelho da marca Melexis, modelo MLX90614, que foi adquirido já soldado em uma placa de circuito integrado contendo alguns componentes básicos para o seu funcionamento, conhecido no meio da eletrônica/Arduino como “Módulo” Figura 12.

Figura 12 - Módulo do sensor de temperatura (MLX90614)

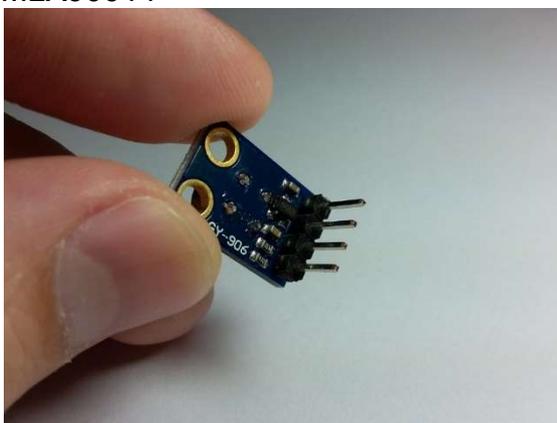


Fonte: Elaborada pelo autor.

Nota-se na Figura 12 que o módulo contém já pronto para uso os terminais com as indicações de pinagem de ligação com outros dispositivos, como o Arduino e também o módulo tem furos para fixação em seu protótipo.

Como o módulo não vem com terminais para ligação direta, foi necessário soldar quatro pinos de uma barra de terminais para que o módulo pudesse ser ligado ao Arduino de modo mais simples e também ser possível uma possível substituição dos fios durante o desenvolvimento do projeto.

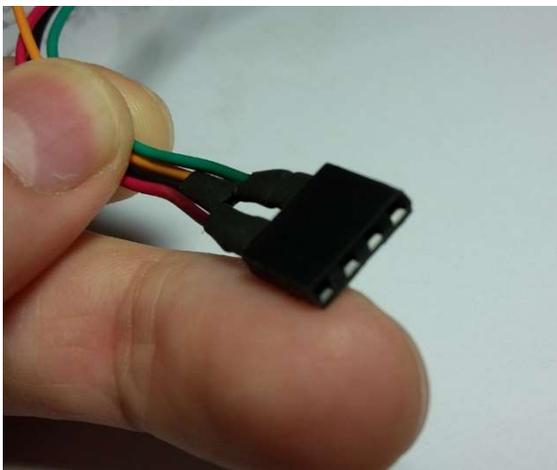
Figura 13 - Parte de trás do módulo MLX90614



Fonte: Elaborada pelo autor.

Na Figura 13 é possível ver os pequenos componentes do módulo já soldados e também a barra de terminais machos que foi adicionada ao módulo com a finalidade de facilitar as conexões dos fios entre o módulo e o Arduino.

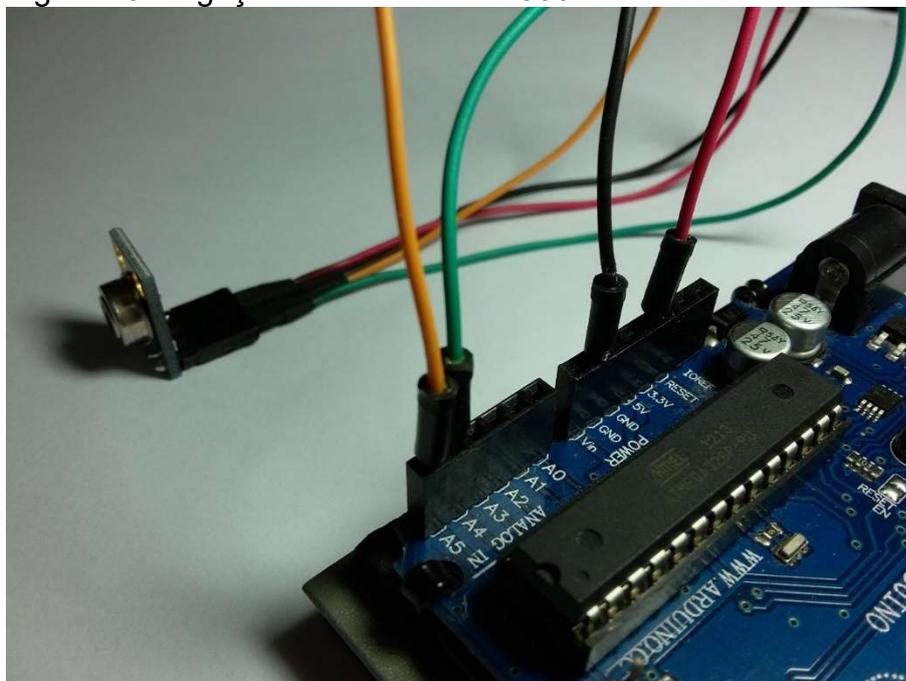
Figura 14 - Conector feito com barra de terminais fêmeas



Fonte: Elaborada pelo autor.

A Figura 14 mostra o cabo feito com uma barra de quatro terminais fêmeas ligando o módulo aos fios.

Figura 15 - Ligação do módulo MLX90614 ao Arduino



Fonte: Elaborada pelo autor.

Com os terminais e fios soldados, as devidas ligações ao Arduino (Figura 15) foram feitas utilizando as portas citadas em tutorial de utilização publicado no site o próprio fabricante. De acordo com os dados publicados, as conexões são as seguintes:

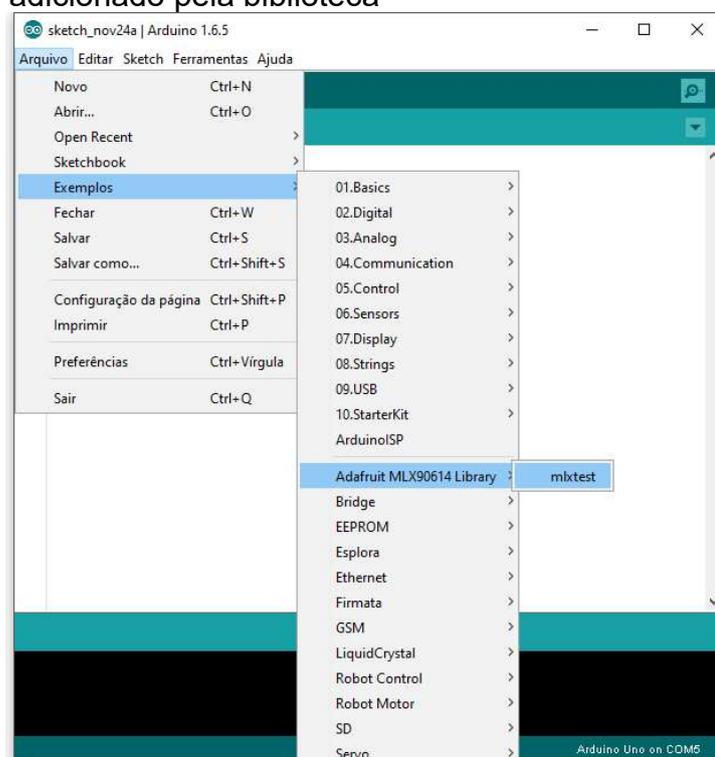
- **GND** no conector negativo do Arduino
- **PWR** no conector positivo (3.3V) do Arduino
- **SDA** no conector I2C **SDA** do Arduino. No modelo UNO é o conector **A4**
- **SCL** no conector I2C clock **SCL** do Arduino. No modelo UNO é o conector **A5**

No mesmo site é possível também baixar a biblioteca necessária para o funcionamento do sensor junto ao Arduino, que nada mais é que os códigos prontos com as funções de programação para facilitar o uso do dispositivo. Está em um arquivo chamado “Adafruit-MLX90614-Library-master.zip”, que foi descompactado e a pasta com os arquivos copiadas para a pasta “libraries” contida no diretório de instalação da IDE do Arduino, onde devem ficar as bibliotecas para serem usadas nos

projetos, caso contrário não é possível a utilização desta no código-fonte. No computador do autor o caminho dos arquivos ficou como: C:\Program Files (x86)\Arduino\libraries\Adafruit-MLX90614-Library-master.

Após copiar a biblioteca descompactada de seu arquivo “.zip” para a pasta “libraries”, ao iniciar a IDE do Arduino uma nova opção é apresentada no menu Arquivo / Exemplos, que é o código de testes do sensor MLX90614 (Figura 16).

Figura 16 - IDE do Arduino com o exemplo adicionado pela biblioteca

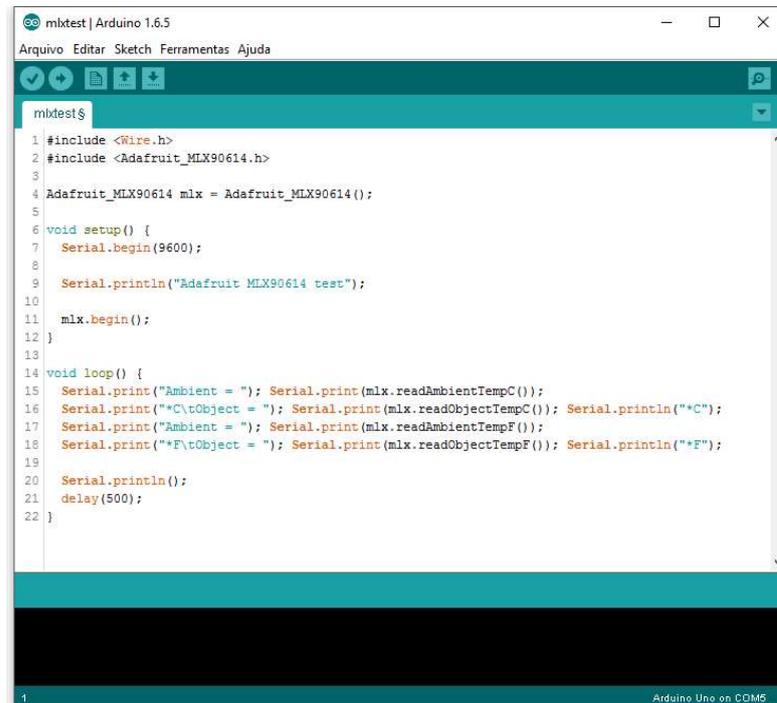


Fonte: Elaborada pelo autor.

Ao clicar no exemplo o código é gerado com as funções disponíveis na biblioteca (Figura 16).

A Figura 17 contém o código básico para o funcionamento do sensor, que foi basicamente comentado na Figura 18 identificando cada função linha por linha.

Figura 17 - Código gerado a partir do exemplo da biblioteca



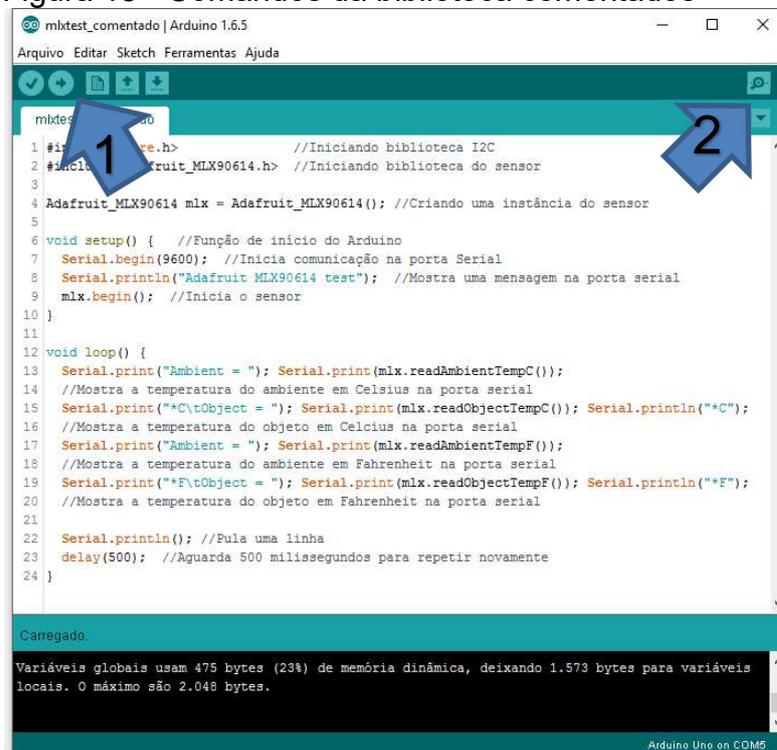
```

1 #include <Wire.h>
2 #include <Adafruit_MLX90614.h>
3
4 Adafruit_MLX90614 mlx = Adafruit_MLX90614();
5
6 void setup() {
7   Serial.begin(9600);
8
9   Serial.println("Adafruit MLX90614 test");
10
11  mlx.begin();
12 }
13
14 void loop() {
15   Serial.print("Ambient = "); Serial.print(mlx.readAmbientTempC());
16   Serial.print("\tObject = "); Serial.print(mlx.readObjectTempC()); Serial.println("\tC");
17   Serial.print("Ambient = "); Serial.print(mlx.readAmbientTempF());
18   Serial.print("\tObject = "); Serial.print(mlx.readObjectTempF()); Serial.println("\tF");
19
20   Serial.println();
21   delay(500);
22 }

```

Fonte: Elaborada pelo autor.

Figura 18 - Comandos da biblioteca comentados



```

1 #include <Wire.h> //Iniciando biblioteca I2C
2 #include <Adafruit_MLX90614.h> //Iniciando biblioteca do sensor
3
4 Adafruit_MLX90614 mlx = Adafruit_MLX90614(); //Criando uma instância do sensor
5
6 void setup() { //Função de início do Arduino
7   Serial.begin(9600); //Inicia comunicação na porta Serial
8   Serial.println("Adafruit MLX90614 test"); //Mostra uma mensagem na porta serial
9   mlx.begin(); //Inicia o sensor
10 }
11
12 void loop() {
13   Serial.print("Ambient = "); Serial.print(mlx.readAmbientTempC());
14   //Mostra a temperatura do ambiente em Celsius na porta serial
15   Serial.print("\tObject = "); Serial.print(mlx.readObjectTempC()); Serial.println("\tC");
16   //Mostra a temperatura do objeto em Celsius na porta serial
17   Serial.print("Ambient = "); Serial.print(mlx.readAmbientTempF());
18   //Mostra a temperatura do ambiente em Fahrenheit na porta serial
19   Serial.print("\tObject = "); Serial.print(mlx.readObjectTempF()); Serial.println("\tF");
20   //Mostra a temperatura do objeto em Fahrenheit na porta serial
21
22   Serial.println(); //Pula uma linha
23   delay(500); //Aguarda 500 milissegundos para repetir novamente
24 }

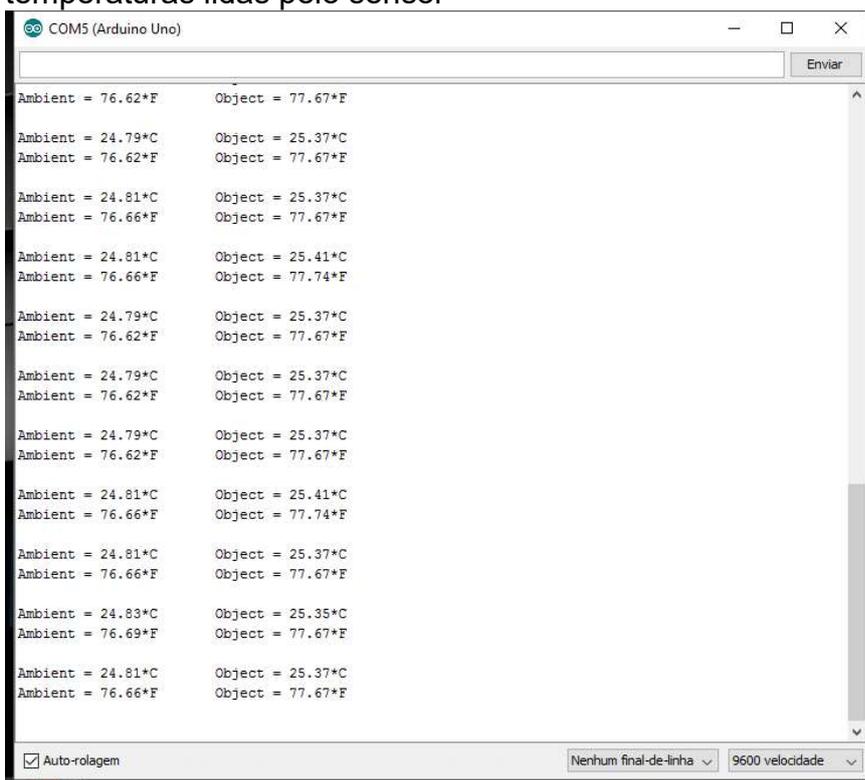
```

Fonte: Elaborada pelo autor.

Após carregar o código exemplo do MLX90614, foi possível fazer o upload do código para o chip do Arduino com um clique no botão (1) denominado “Carregar”

marcado na Figura 18, e depois de que nenhuma mensagem de erro foi mostrada abaixo, com um clique no botão (2) denominado “Monitor Serial”, a janela é aberta para a visualização das mensagens enviadas pelo Arduino via porta serial utilizando o comando `Serial.print()` que pode se ver no código-fonte da Figura 19.

Figura 19 - Janela do "Monitor Serial" mostrando as temperaturas lidas pelo sensor



The screenshot shows the Serial Monitor window for COM5 (Arduino Uno). The window displays a series of temperature readings for two sensors: Ambient and Object. The readings are shown in pairs, with the Ambient temperature on the left and the Object temperature on the right. The temperatures are displayed in both Celsius and Fahrenheit. The window also includes a scroll bar, a 'Enviar' button, and a status bar at the bottom with options for 'Auto-rolagem', 'Nenhum final-de-linha', and '9600 velocidade'.

```
COM5 (Arduino Uno)
Enviar
Ambient = 76.62°F    Object = 77.67°F
Ambient = 24.79°C    Object = 25.37°C
Ambient = 76.62°F    Object = 77.67°F
Ambient = 24.81°C    Object = 25.37°C
Ambient = 76.66°F    Object = 77.67°F
Ambient = 24.81°C    Object = 25.41°C
Ambient = 76.66°F    Object = 77.74°F
Ambient = 24.79°C    Object = 25.37°C
Ambient = 76.62°F    Object = 77.67°F
Ambient = 24.79°C    Object = 25.37°C
Ambient = 76.62°F    Object = 77.67°F
Ambient = 24.79°C    Object = 25.37°C
Ambient = 76.62°F    Object = 77.67°F
Ambient = 24.81°C    Object = 25.41°C
Ambient = 76.66°F    Object = 77.74°F
Ambient = 24.81°C    Object = 25.37°C
Ambient = 76.66°F    Object = 77.67°F
Ambient = 24.83°C    Object = 25.35°C
Ambient = 76.69°F    Object = 77.67°F
Ambient = 24.81°C    Object = 25.37°C
Ambient = 76.66°F    Object = 77.67°F
```

Auto-rolagem Nenhum final-de-linha 9600 velocidade

Fonte: Elaborada pelo autor.

Na janela do Monitor Serial, são exibidas as mensagens de acordo com o código-fonte do exemplo carregado (Figura 19), onde é mostrado a temperatura ambiente, e do objeto à frente do sensor MLX90614 em Celsius, e logo na próxima linha em Fahrenheit, e assim sucessivamente fica repetindo indefinidamente, tal como foi programado seguindo o código-fonte.

Após esses passos, se teve o funcionamento do sensor MLX90614, e foi possível iniciar alguns testes quanto à sua eficácia e precisão a seguir.

Para uma boa medição e testes do sensor, foi adquirido também um dispositivo conhecido como “Medidor de temperatura a laser”, que nada mais é que um sensor de temperatura infravermelha com uma “mira à laser”, que pode ser visto na Figura 20 e foi utilizado como referência para as medições realizadas.

Figura 20 - Medidor de temperatura "à laser"



Fonte: Elaborada pelo autor.

O Medidor de temperatura adquirido possui as seguintes características:

Tabela 4 - Especificações do medidor de referência

Faixa de temperatura	-50 ~ 330°C (-58 ~ 626°F)
Precisão	$\pm 1.5^{\circ}\text{C}$ de 0°C ~ 330°C ou 1,5% $\pm 3^{\circ}\text{C}$ de -50°C ~ 0°C
Resolução	0.1°C ou 0.1°F
Tempo de resposta	500ms, 95%
Resposta espectral	4-14 μm
Distância para o tamanho do ponto	12:1

Fonte: Elaborada pelo autor.

Para confirmar o equipamento interno do medidor utilizado como referência, foi desmontado o dispositivo para uma rápida análise de seus componentes:

Figura 21 - Lente do termômetro comercial



Fonte: Elaborada pelo autor.

Na imagem do dispositivo desmontado Figura 21, é possível ver que ele possui como componente principal para a medição da temperatura um medidor infravermelho, de tamanho menor que o utilizado neste trabalho (MLX90614), que não foi possível confirmar o modelo específico do sensor do dispositivo, pois não há referências nem no manual de instruções nem do fabricante, e nem em pesquisas na internet.

Figura 22 - Lente do termômetro comercial



Fonte: Elaborada pelo autor.

Na Figura 22 vemos uma lente que fica à frente do sensor comercial, do tipo lente de Fresnel, que é uma lente que permite maior passagem de luz, deixando a luz visível a distâncias maiores.

Os dois servo motores utilizados nesse trabalho foram do modelo Micro Servo 9g SG90 marca Tower Pro™, e (Figura 23) foram escolhidos por seu tamanho reduzido, leves, precisão razoável e baixo custo. Um ponto importante é o seu peso, pois a disposição da montagem do projeto é importante que o peso seja o mínimo possível.

Figura 23 - Servo motor utilizado neste trabalho



Fonte: Elaborada pelo autor.

O servo motor foi ligado ao Arduino, de acordo com esquema mostrado no próprio site do Arduino em <https://www.arduino.cc/en/Tutorial/Sweep>, ligando seus 3 terminais: Positivo, negativo e sinal, onde a porta de sinal foi a porta 09 do Arduino.

Na Figura 24 contém o código-fonte básico fornecido pelo IDE do Arduino para o funcionamento ou teste do servo-motor, que faz com que ele movimente de um lado para o outro ininterruptamente, acessado pelo menu Arquivo / Exemplos / Servo / Sweep.

O código fonte “Sweep” mostrado na Figura 24, faz com que o servo motor faça como um movimento de vai e vem (meia volta vai, meia volta vem), demonstrando seu funcionamento.

Após o teste de todos os componentes físicos do projeto, foi dado início à montagem do protótipo. A ligação completa dos componentes no Arduino ficaram como citado na Tabela 5.

Figura 24 - Código teste do servo motor comentado

```

1 #include <Servo.h>
2 |
3 Servo myservo; // Cria o objeto servo motor
4
5 int pos = 0; // variável que armazena a posição do servo motor
6
7 void setup()
8 {
9   myservo.attach(9); // porta onde está conectado o servo motor
10 }
11
12 void loop()
13 {
14   for(pos = 0; pos <= 180; pos += 1) // vai de 0 a 180 graus de 1 a 1 grau
15   {
16     myservo.write(pos); // indica a posição para o servo
17     delay(15); // aguarda 15ms para o servo chegar a posição
18   }
19   for(pos = 180; pos>=0; pos-=1) // vai de 180 a 0 graus de 1 a 1 grau
20   {
21     myservo.write(pos); // indica a posição para o servo
22     delay(15); // aguarda 15ms para o servo chegar a posição
23   }
24 }
25

```

Salvo.

2 Arduino Uno on COM5

Fonte: Elaborada pelo autor.

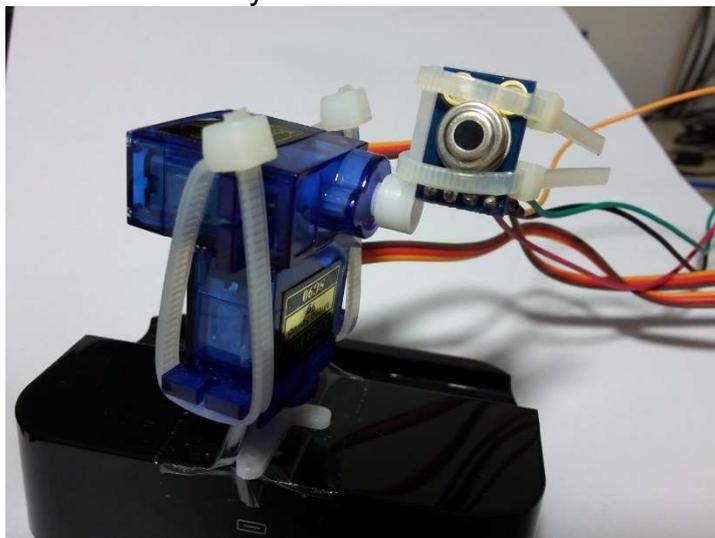
Tabela 5 - Ligação dos componentes no Arduino

Componente	Porta no componente	Porta do Arduino
Sensor MLX90614	GND Vin SCL SDA	Energia Negativo (-) Energia Positivo (+) 3.3V Analogico 5 (A5) Analogico 4 (A4)
Servo Motor 1	GND POWER Sinal	Energia Negativo (-) Energia Positivo (+) 5V Porta digital 2
Servo Motor 2	GND POWER Sinal	Energia Negativo (-) Energia Positivo (+) 5V Porta digital 3

Fonte: Elaborada pelo autor.

Terminadas as ligações, foi montado os motores (Figura 25) e posicionado o sensor de temperatura em um dos eixos (eixo y), onde o servo motor inferior controla a movimentação horizontal (eixo x) e o servo motor superior controla a movimentação vertical (eixo y).

Figura 25 - Servo motores montados para fazer a varredura de x e y



Fonte: Elaborada pelo autor.

Os servo motores foram presos à uma base com peso interno para não se movimentar durante a varredura dos pontos, e os motores presos entre si por abraçadeiras de plástico.

5 TESTES E RESULTADOS

Com os componentes montados e ligados de maneira correta, foi iniciado alguns testes de posição dos motores e principalmente do eixo onde fica o sensor e também do código-fonte para a criação do programa para enviar ao Arduino. Segue abaixo na Figura 26 o código-fonte comentado explicando cada comando utilizado para o funcionamento.

Figura 26 - Código fonte completo e comentado da solução encontrada para o Arduino

```

TCC_ARDUINO_comentado | Arduino 1.6.5
Arquivo Editar Sketch Ferramentas Ajuda

TCC_ARDUINO_comentado
1 #include <Servo.h>
2 #include <Wire.h>
3 #include <Adafruit_MLX90614.h>
4
5 Adafruit_MLX90614 sensor = Adafruit_MLX90614();
6
7 Servo servox, // criação de objetos para controlar os servos
8   servoy;
9
10 int posicao_x = 0; //inicialização de variáveis de posição dos servos
11 int posicao_y = 0;
12
13 int total_x = 10; //ângulos de "visão" do objeto
14 int total_y = 10;
15
16 void setup()
17 {
18   servox.attach(2); // servo x "horizontal"
19   servoy.attach(3); // servo y "vertical"
20   Serial.begin(9600); // inicia porta serial a 9600 bits/s
21   sensor.begin(); // inicia o sensor
22   delay(300); // aguarda 300ms para iniciar
23 }
24
25 void loop()
26 {
27   if (Serial.available()) {
28     char recebido;
29     recebido = Serial.read();
30     if (recebido == 'i') {
31       char buffer[15]; // para colocar zeros à esquerda...
32       if (posicao_x != total_x+1) {
33         for(posicao_x = 0; posicao_x <= total_x; posicao_x += 1) // de x=0 até x=total_x
34         {
35           servox.write(posicao_x);
36           delay(200);
37           for(posicao_y = 0; posicao_y <= total_y; posicao_y += 1) // y...
38           {
39             servoy.write(posicao_y); // indica a posição para o servo
40             delay(200); // espera 15ms para o servo alcançar a posição pedida
41             sprintf(buffer, "%02d,%02d", posicao_x, posicao_y); // envia para serial "valorX, valorY,"
42             Serial.print(buffer);
43             Serial.print(sensor.readObjectTempC());
44             delay(400);
45             Serial.println();
46           } //fim for y
47         } //fim for x
48       } //fim if
49     } //fim if recebido
50   } //fim if serial available
51 }
52

```

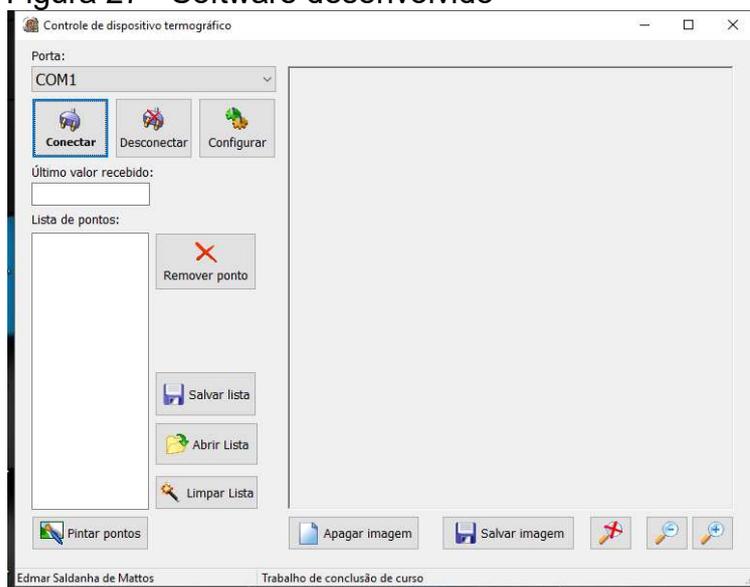
Fonte: Elaborada pelo autor.

O código criado para o Arduino se mostrou totalmente funcional para o início do desenvolvimento do software de recebimento dos dados enviados via porta serial que foi desenvolvido para criar a imagem termográfica, pois o Arduino se tornou capaz de fazer as seguintes funções:

- Iniciar a varredura após a leitura de comando específico;
- Varrer os pontos horizontais e verticais (x e y) e ler a temperatura de cada ponto;
- Enviar os dados via porta serial para o computador no formato desejado inicialmente do trabalho (posiçãoX, posiçãoY, temperatura);
- Após a varredura total dos pontos, o dispositivo encerra seu funcionamento.

O software desenvolvido foi criado baseando-se na premissa de fazê-lo de maneira mais simples possível, mas também com maior controle sobre o funcionamento do processo, onde foi possível ver o que está acontecendo o tempo todo.

Figura 27 - Software desenvolvido



Fonte: Elaborada pelo autor.

Desenvolvido utilizando o Delphi 2010, o software foi desenvolvido após a conclusão da montagem do protótipo do Arduino. Na Figura 27, está o software com todas as funções implementadas para o controle e para a captura dos dados enviados via porta serial do protótipo, que são as temperaturas e os pontos correspondentes (x e y).

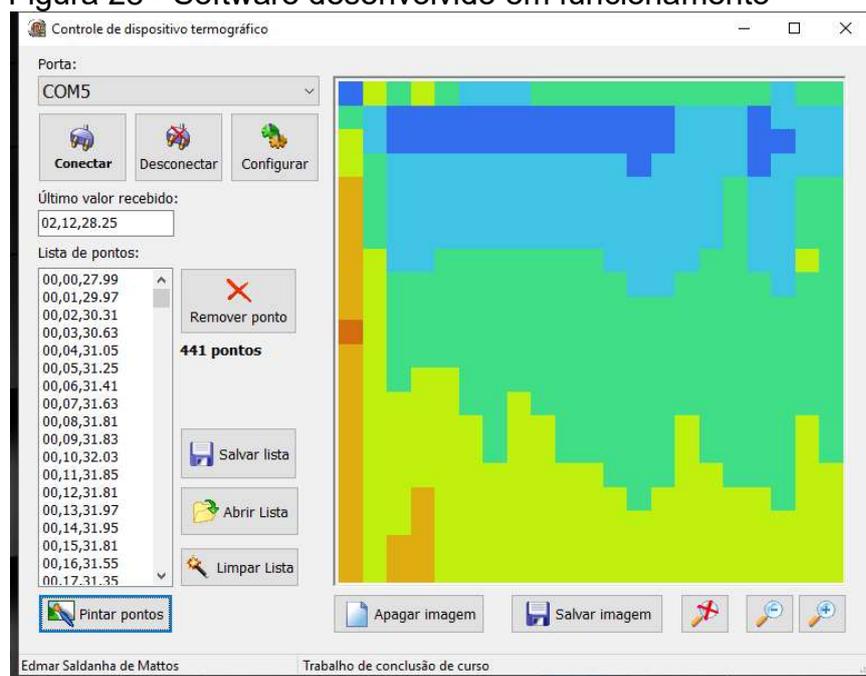
O software desenvolvido é capaz de conectar à porta serial onde o Arduino está ligada ao computador, e depois começa a receber os dados enviados por ele com os

pontos capturados pelo sensor de temperatura e é mostrado em tempo real no campo “Último valor recebido” (Figura 27), também ele é adicionado à “Lista de pontos”, onde ficam todos os pontos carregados a partir do Arduino. A lista de pontos pode ser salva em um arquivo de texto, como também pode ser carregada a partir de um arquivo e após, é possível mostrar o desenho capturado pelos pontos ao clicar no botão “Pintar pontos”. Também o software tem as seguintes características:

- Opção para conectar em qualquer porta serial onde o dispositivo esteja plugado e também é possível configurar a porta serial de acordo com necessidade específica;
- Cria uma lista de pontos onde é possível visualizar a medição enquanto o dispositivo varre o objeto;
- Pode remover pontos que possam ser lidos errados;
- Salva a lista de pontos em um arquivo de texto para uma futura análise;
- Carrega lista de pontos previamente salva pelo próprio programa;
- É capaz de dar zoom na imagem gerada.

No desenvolvimento do software somente foi utilizado um componente externo, que não é do próprio IDE do Delphi 2010, que foi o componente TComPort, pois sua utilização facilita o processo de captura dos dados da porta serial.

Figura 28 - Software desenvolvido em funcionamento



Fonte: Elaborada pelo autor.

Na Figura 28 pode ser visto uma varredura completa, com 441 pontos capturados, e é possível ver a imagem gerada, que também está com o zoom aumentado, pois possui poucos pontos (21x21 pontos), se pode notar que há grande quantidade de ruído na imagem, onde a parte de baixo é a mais quente e a parte de cima é a mais fria.

Apesar de ser possível notar uma pequena variação da temperatura na imagem gerada, que faz com que haja alteração nas cores da imagem, a temperatura pode não corresponder a real temperatura do objeto, como podemos ver abaixo na tabela a seguir feita medindo um cubo de gelo.

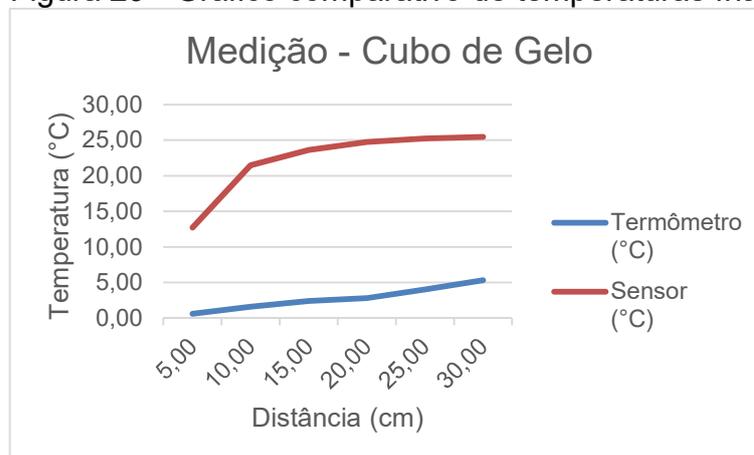
Tabela 6 - Comparativo de temperaturas frias

<i>Medição - Cubo de Gelo</i>		
Distância (cm)	Termômetro (°C)	Sensor (°C)
5,00	0,60	12,73
10,00	1,60	21,49
15,00	2,40	23,65
20,00	2,80	24,75
25,00	4,00	25,23
30,00	5,30	25,49

Fonte: Elaborada pelo autor.

De acordo com as medições feitas com o termômetro comercial foi possível notar a grande diferença de temperatura do sensor para a temperatura real medida. O sensor no mínimo teve uma taxa de erro de mais de 10 graus Celsius de diferença com a temperatura real para um objeto gelado.

Figura 29 - Gráfico comparativo de temperaturas frias



Fonte: Elaborada pelo autor.

Na Figura 29 temos visualmente a curva de temperatura medida entre o termômetro e o sensor MLX90614, onde é possível observar que o sensor não teve um rendimento satisfatório para a medição em distância sem um conjunto mais elaborado de sensor e lentes óticas adequadas.

Vemos na Figura 29 que a diferença é muito grande, e também não seguiu a variação correta mesmo em distâncias diferente, tendo uma curva muito errônea em relação ao termômetro.

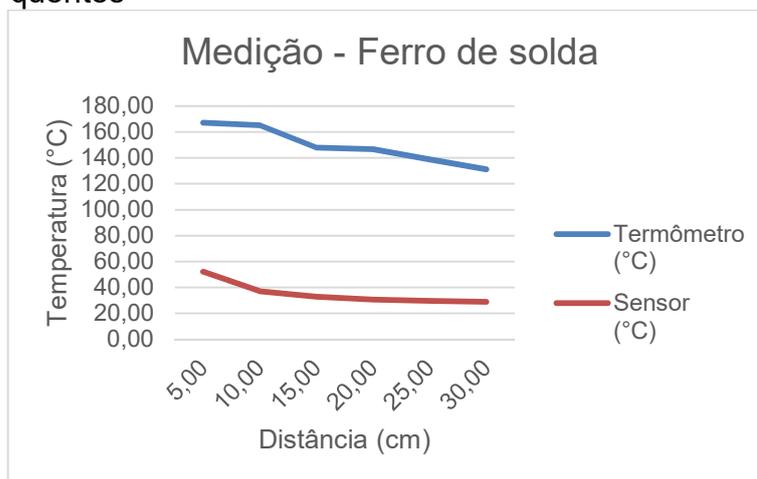
A seguir os dados medidos em temperaturas altas:

Tabela 7 - Comparativo de temperaturas quentes

Medição - Ferro de solda		
Distância (cm)	Termômetro (°C)	Sensor (°C)
5,00	167,08	52,03
10,00	165,30	37,10
15,00	148,00	32,81
20,00	146,60	30,79
25,00	138,80	29,83
30,00	131,20	28,95

Fonte: Elaborada pelo autor.

Figura 30 - Gráfico comparativo de temperaturas quentes



Fonte: Elaborada pelo autor.

Como se pode perceber na Tabela 7 e Figura 30 o erro de valores também persiste e a margem de erro é de mais de 100 graus Celsius de diferença para objetos quentes, sendo impossível de haver uma medição correta com essa grande variação.

6 CONSIDERAÇÕES FINAIS

No início do trabalho, o sensor MLX90614 mostrou-se o componente ideal para o desenvolvimento do dispositivo, apesar de poucas informações sobre o seu funcionamento, e pouca informação sobre o termografia, foi possível identificar que um sensor infravermelho deste modelo ou muito semelhante é o ideal para a criação de um termógrafo funcional.

Como não foi possível o acoplamento de uma lente semelhante à do termômetro “laser” pois não há no mercado fácil acesso a esse tipo de lente, e também no tamanho indicado e para esta aplicação específica carece de algumas peculiaridades como tamanho, peso e preço, fez com que não fosse inserido no projeto.

Os servo motores foram razoáveis em seu papel nos resultados do trabalho, pois eles têm alguns problemas que podem ser notados durante os testes deste trabalho, como alguns ângulos incorretos, também por algumas vezes ele não responde aos comandos corretamente e pula um ângulo e adianta outro logo em seguida, fazendo com que aumente o ruído da imagem e também diminua sua confiabilidade.

O Arduino mostrou-se satisfatório para o desenvolvimento deste trabalho, pois sua facilidade de desenvolvimento, baixo custo, vasta quantidade de exemplos e bibliotecas prontas para uso, demonstram que ele tem sempre um grande potencial para o desenvolvimento de trabalhos de baixo e média complexidades sem comprometimento de confiabilidade.

O software desenvolvido tem grande potencial para funcionar com uma melhor versão ou mesmo um outro dispositivo que venha a ser desenvolvido utilizando a mesma técnica de extração de temperatura por varredura de pontos, pois o software está apto a receber dados no mesmo formato e processá-los de maneira confiável e funcional.

O desenvolvimento utilizando o Delphi 2010, mostrou-se rápido e fácil de aplicar os conhecimentos adquiridos ao longo da formação acadêmica, e também muito rico em componentes visuais, sendo útil para o desenvolvimento de aplicações visuais e pesquisas acadêmicas.

TRABALHOS FUTUROS

No decorrer do trabalho, e principalmente no fim do desenvolvimento desde, foi possível identificar vários aspectos que podem ser melhorados em trabalhos futuros, como também em novas adaptações desde mesmo trabalho, citados a seguir.

Quanto ao sensor:

O sensor MLX90614, como já citado, apesar de não ter obtido um desempenho satisfatório neste trabalho, pode ser utilizado em conjunto com algumas lentes especiais, ou mesmo comprado junto de uma lente que seja possível uma leitura mais próxima da leitura real do objeto a ser medido.

Também, outras sugestões podem ser observadas, como a adição de espelhos curvos para a concentração da radiação infravermelha no captor do sensor poderia melhorar sua captação e conseqüentemente melhorar seu desempenho.

Outras ideias podem ser incorporadas, como outros sensores para uma média de temperatura entre eles para aumentar a precisão, ou mesmo uma matriz de vários sensores poderiam formar uma imagem em tempo real com a temperatura medida, assim como os termógrafos profissionais de alto custo.

Quanto aos motores:

Os servo motores cumpriram seu papel em partes, pois um ponto crucial do trabalho era a precisão dos movimentos, e nesse quesito o servo motor não teve um resultado tão satisfatório, quanto poderia ser se fosse adotado um motor de passo por exemplo. Com isso provavelmente a questão de precisão nos movimentos melhoraria bastante, apesar de aumentar a complexidade pois os motores de passo precisariam de alguns componentes a mais e maiores regulagens quanto ao ângulo de varredura.

Quanto ao software:

Apesar do software ter suas funcionalidades inicialmente planejadas todas implementadas, alguns componentes utilizados não tem grande flexibilidade para o desenvolvimento, como o componente TComPort, que faltam algumas funções específicas para monitoramento mais detalhado da porta serial e dos dados enviados. Também o componente TImage do Delphi não tem funções tão completas, que podem travar o desenvolvimento de algumas funções desejadas no programa desenvolvido, fazendo com que algumas adaptações tenham sido feitas.

REFERÊNCIAS

- ARDUINO. **Introduction Arduino**, 2015. Disponível em: <<http://www.arduino.cc/en/Guide/Introduction>>. Acesso em: 05 Jun. 2015. Tradução nossa.
- ARDUINOBOARDUNO. **Arduino**, 2015. Disponível em: <<http://www.arduino.cc/en/Main/ArduinoBoardUno>>. Acesso em: 05 Jun. 2015.
- CANTÙ, M. **Dominando o Delphi 2005 "A Bíblia"**. 3ª. ed. São Paulo: Pearson Prentice Hall, 2006.
- CARVALHO, T. **Espectro Eletromagnético**. InfoEscola, 2006-2015. Disponível em: <<http://www.infoescola.com/fisica/espectro-eletromagnetico/>>. Acesso em: 05 Jun. 2015.
- CHIODINI, G. Gchiodini's Blog, 2015. Disponível em: <<https://gchiodini.wordpress.com/termografia/>>. Acesso em: 05 Jun. 2015.
- AZEVEDO, E.; CONCI, A.; LETA, F. R. **Computação Gráfica - Teoria e Prática**. 2ª. ed. Rio de Janeiro: Elsevier Editora Ltda, v. 2, 2008.
- CRNILA, D. **ComPort Library**. **SourceForge**, 2003. Disponível em: <<http://sourceforge.net/projects/comport/>>. Acesso em: 01 out. 2015.
- EMBARCADERO Delphi. **Wikipédia**, 2015. Disponível em: <http://pt.wikipedia.org/wiki/Embarcadero_Delphi>. Acesso em: 05 Junho 2015.
- EVANS, M.; HOCHENBAUM, J.; NOBLE, J. **Arduino em ação**. São Paulo: Novatec Editora Ltda., 2014.
- FERNANDES, A. D. A. Temperatura da pele durante o exercício: comparação de métodos. Universidade Federal de Viçosa. Viçosa, MG. 2013.
- FLIR T660 Infrared Camera. SHOPFLIR, 2015. Disponível em: <<http://www.shopflir.com/product/flir-t660-infrared-camera>>. Acesso em: 05 Jun. 2015.
- GASPAR, A. **Física - Ondas - Óptica - Termodinâmica**. 2ª. ed. São Paulo: Ática, v. 2, 2002.
- GIL, A. C. **Como Elaborar Projetos de Pesquisa**. 4ª. ed. São Paulo: Atlas, 2002.
- GONZALES, R. C.; WOODS, R. E. **Processamento Digital de Imagens**. 3ª. ed. São Paulo: Pearson Prentice Hall, 2010.
- HECHT, E. **Óptica**. 2ª. ed. [S.l.]: Gráfica de Coimbra,Lda., 2002.

HIMARA, K. **Engenharia de Software - Qualidade e Produtividade com Tecnologia**. Rio de Janeiro: Elsevier, 2012.

INFRATHERM. A TERMOGRAFIA. Infratherm Termografia, 2010. Disponível em: <<http://www.infratherm.net.br/termografia>>. Acesso em: 20 Mar. 2015.

LOBO, E. J. R. **Curso de Engenharia de Software**. São Paulo: Digerati Books, 2008.

LIMA, C. R. C.; MARCORIN, W. R. **Análise dos Custos de Manutenção**. Revista de Ciência & Tecnologia, Santa Bárbara d'Oeste, 18 Dez. 2003. 39 p.

MENDONÇA, L. V. **Termografia por infravermelhos** - Inspeção de betão. Engenharia & Vida, Set. 2005. 53-57 p.

MORAIS, T. C. M. D.; MOURA, D. F. P. D. Adequação das conexões em subestações para redução de pontos quentes. Foz do Iguaçu, 2014. 15 p.

MOURA, D. F. P. D.; MORAIS, T. C. M. D. **Adequação das conexões em subestações para redução de pontos quentes**. Foz do Iguaçu, p. 15. 2014.

PEREIRA, J. A.; SANTOS, G. L. Utilização da análise de componentes principais em termografia. **Revista multidisciplinar da Uniesp**, Dez. 2010.

WIREMAN, T. **Developing Performance Indicators for Managing Maintenance**. 2ª. ed. New York: Industrial Press, 2005.