

Projeto de um Aplicativo Desktop no App Designer para Calibração de Sensor TCS230 com Arduino

Josiele Lima Barreto
Universidade Federal do Oeste do
Pará (Ufopa)
Santarém, Pará, Brasil
josielebarreto123@gmail.com

Gilson Braga Junior
Universidade Federal do Oeste do
Pará (Ufopa)
Santarém, Pará, Brasil
gilson.braga@ufopa.edu.br

ABSTRACT

Calibration is fundamental for the proper functioning of a sensor, considering its sensitivity and susceptibility to influences from specific processes in which it is employed. This study aims to develop a desktop application to assist in calibrating the TCS230 color sensor using MATLAB® software through the App Designer tool. This application is integrated into a prototype of a machine for sorting objects by color, with data input received from an Arduino board. Three different tests were conducted, involving the use and non-use of the application and variations in ambient lighting conditions around the prototype. As a result, a user manual was generated, and the calibration process was streamlined through the implementation of a calibration button, which yielded frequency interval values of maximum and minimum ranges. Consequently, this work demonstrates that the platform is efficient and applicable in various scenarios focused on color identification using a TCS230-type sensor.

KEYWORDS

Sensor de cor; Arduino; MATLAB; App Designer.

1 INTRODUÇÃO

A automação é um elemento indispensável quando o objetivo é fornecer um produto ou serviço o mais confiável e eficiente possível [1]. Com o progresso da automação, o uso de tecnologias de sensores e microcontroladores se tornou amplamente difundido. O microcontrolador é empregado como uma ferramenta que auxilia no trabalho humano. Quanto aos sensores, eles se destacam devido à sua eficácia em obter informações sobre variáveis físicas do ambiente ao seu redor. Essa habilidade em adquirir dados e facilitar o armazenamento e processamento é fundamental para visualização e análise posterior [2].

Dentre as aplicações da automação, destaca-se a classificação de objetos. Vários estudos têm se concentrado na implementação de sistemas automatizados de classificação, com ênfase na detecção e diferenciação de cores. Essas pesquisas abrangem setores como a classificação de grãos, frutas, plantas, e em laboratórios químicos. Como exemplificado em Maia [3] em que o autor criou um protótipo para identificar e separar cores de grãos de café usando um sensor RGB TCS34725. Em Silva [4], é mostrado o desenvolvimento de um sistema automatizado com

Arduino e o sensor TCS230 para coletar e controlar variáveis de cor, associando-as à maturidade e qualidade da goiaba. E em Souza [5] buscou-se desenvolver um protótipo para monitorar de forma contínua a urina humana através de um sistema de detecção de variações da urina utilizando uma plataforma ESP32 e de sensores (pH e reconhecimento de cores – TCS230).

Esses estudos são essenciais, pois abordam setores nos quais a precisão na identificação de cores desempenha papel crucial em processos de seleção e qualidade. Considerando a limitação humana nesses processos, que raramente ocorre quando máquinas são empregadas, pois elas têm a capacidade de estabelecer um padrão uniforme visualmente e diminuir o tempo gasto na execução dessas tarefas. No entanto, a precisão e confiabilidade das medições realizadas por esses dispositivos dependem diretamente de uma calibração adequada. A calibração compara as leituras fornecidas pelo sensor com um padrão de referência confiável, determinando e corrigindo eventuais desvios ou erros de medição.

Trabalhos anteriores abordaram os desafios associados à calibração de sensores de cor, reconhecendo a necessidade de garantir a consistência e confiabilidade das medições. Em Souza [5], o autor destaca a importância da calibração para a obtenção de dados mais confiáveis e íntegros em seu trabalho. Já em Oliveira et al. [6] os autores utilizaram a metodologia de calibração para verificar a confiabilidade da leitura das cédulas monetárias utilizadas na pesquisa.

No cenário específico deste estudo, destaca-se a pesquisa de Da Silva [7], que desenvolveu uma máquina separadora de peças por cor, incorporando o módulo sensor de cor TCS230. Este trabalho serviu de inspiração para a presente pesquisa, focada no desenvolvimento de um aplicativo desktop para auxiliar na calibração desse módulo sensor. A pesquisa visa proporcionar comodidade e rapidez durante o processo de calibração, uma vez que sensores desse tipo podem ser influenciados por diferentes condições de iluminação, reflexões, sombras e variações nos objetos, resultando em medições imprecisas. O uso do protótipo em outros ambientes pode acarretar variações no envio de dados do sensor, assim como a troca do componente, já que cada sensor TCS230 pode apresentar variações de sensibilidade em diferentes regiões do espectro de cores. A calibração permite ajustar essas variações, garantindo uma resposta mais confiável e consistente do sensor em todas as faixas de cores.

2 FUNDAMENTAÇÃO TEÓRICA

A automação tem sido uma das principais forças impulsionadoras do progresso tecnológico nas últimas décadas. Através da aplicação de máquinas, inteligência artificial e robótica, a automação tem revolucionado setores industriais, comerciais e até mesmo domésticos. É uma área que engloba desde processos simples até processos de sistemas complexos que envolvem máquinas, software e algoritmos avançados. Tem como objetivo aumentar a eficiência, a produtividade e a precisão, além de reduzir o tempo e o esforço humano necessários para execução de determinadas atividades [1].

Ao estudar automação, como sistemas industriais, comerciais, automotivos, entre outros, é necessário determinar as variáveis do sistema. Nesse contexto, é preciso obter o valor de uma variável física no ambiente monitorado, e esse é o trabalho do sensor. Um sensor pode ser definido como um dispositivo sensível que responde a alguma forma de energia em um sistema (como cinética ou luminosa) e tem a capacidade de mensurar certas quantidades do sistema, como temperatura, posição e velocidade [8].

A evolução constante da tecnologia tem impulsionado o desenvolvimento de sensores cada vez mais avançados e precisos, abrindo novas fronteiras em campos como a automação industrial, a medicina, a pesquisa espacial e a Internet das Coisas (IoT). Existem diversos tipos de sensores, categorizados com base em suas principais aplicações e princípios de funcionamento: sensores de temperatura, de nível, de torque, de pressão, de umidade, de luz, entre outros. A escolha do sensor apropriado depende das necessidades da aplicação, das condições ambientais e dos requisitos de precisão.

Para classificação de objetos no contexto das cores são utilizados sensores de cor.

Os sensores de cor são sensores baseados no efeito fotoelétrico. Seu funcionamento consiste na emissão de uma luz branca, que ao atingir a superfície em análise retorna refletida. Como resultado dessa reflexão, é produzido um padrão RGB (*Red, Blue e Green*), que permite ao sensor emitir sinais discretos ou de comunicação em termos de especificações de cores que caracterizam o objeto sob inspeção [7].

Existem diferentes tipos de sensores de cor, cada um com seus princípios de funcionamento e características específicas. Alguns exemplos comuns incluem: Sensores RGB, espectrofotométricos, e filtro interferencial. Alguns sensores de cor populares disponíveis no mercado são desenvolvidos em formato de módulos, como o TCS230, TCS3200, TCS34725 e APDS-9930 e por apresentar um custo acessível, podem ser usados didaticamente.

2.1 TCS230

O sensor de cor TCS230, mostrado na Figura 1, é um sensor de cor RGB desenvolvido pela AMS (Antiga TAOS Inc.). Trata-se de um sensor óptico que pode detectar e medir cores com base na intensidade da luz em diferentes faixas do espectro, possui 4 LEDs (*Light Emitting Diode* – Diodo Emissor de Luz),

capacitores, resistores, 8 pinos e um componente CMOS com uma matriz 8x8 de fotodiodos [9].



Figura 1: Módulo sensor de cor RGB TCS230 [10].

De acordo com Da Silva [3], o módulo TCS230, tem seu princípio de funcionamento baseado na conversão da cor da luz em frequência, filtrando os dados RGB da fonte de luz e transformando em uma onda quadrada (50% *duty cycle* – ciclo de trabalho) com frequência diretamente proporcional à intensidade da luz, isto é, a onda apresenta metade do período em nível alto e a outra metade em nível baixo. Existem 64 fotodiodos, 16 com filtros vermelhos, 16 com filtros azuis, 16 com filtros verdes e 16 sem filtros. Esses quatro fotodiodos são interligados para minimizar os efeitos do brilho desigual da luz incidente. Todos os fotodiodos da mesma cor são conectados em paralelo e o fotodiodo que o dispositivo usa durante a operação pode ser selecionado por pinos [11].

Considerando a teoria de cores RGB, existe três valores distribuídos, o primeiro para R (vermelho), o segundo para G (verde) e o terceiro para B (azul) que são combinados com diferentes intensidades da cor (sem filtro), para obter tons diferentes. As medições são feitas da seguinte forma: Quatro LEDs brancos iluminam o objeto a ser identificado. A luz refletida é, então, enviada para o sensor TCS230. As informações obtidas são enviadas para um microcontrolador, onde um software é utilizado para medir as respectivas frequências dos canais R, G e B. Isso significa que existem três valores diferentes, um para cada variável [12].

O sensor de cor TCS230 é comumente utilizado em aplicações como detecção de cores, controle de iluminação, classificação de objetos, robótica e automação industrial. Sua simplicidade de uso o torna uma escolha popular em diversas aplicações. Este módulo possui entradas e saídas digitais que podem ser conectadas diretamente a um microcontrolador.

2.2 Placa Arduino

Para obter as informações dos sensores e controlar os atuadores, pode-se utilizar o Arduino para prototipagem. O Arduino é uma plataforma de hardware e software livres, isto é, pode ser utilizado por qualquer pessoa com a liberdade de estudar, adaptar/modificar e distribuir.

A parte de hardware consiste em uma placa com microcontrolador, que pode ser considerado como um pequeno

computador, já que é composto por microprocessador, memória RAM, memória flash, temporizadores, contadores, dentre outras.

O software é a IDE (*Integrated Development Environment*) do Arduino, que permite que o usuário escreva um programa de computador baseada na linguagem C compreendida pelo Arduino. Esse programa executado no computador é denominado de sketch, que fará o envio para a placa Arduino [13].

Conforme discutido por McRoberts [13], a placa Arduino pode ser utilizado juntamente com qualquer dispositivo que emita dados ou possa ser controlado (LEDs, sensores, receptores GPS, módulos Ethernet), o que permite adquirir informações e ser sensível ao ambiente ao seu redor fazendo o controle de luzes, motores e outros atuadores.

2.3 Matlab: Ferramenta *App Designer*

O MATLAB® (*MATRIX LABORATORY* - Laboratório de Matrizes) é um software que oferece uma ampla gama de recursos, sendo um deles o suporte à comunicação serial e programação, recursos gráficos para criar interfaces de usuário personalizadas, permitindo interações com o Arduino para controle, monitoramento e análise de dados em tempo real.

A ferramenta *App Designer* (Designer de Aplicativos) do MATLAB®, é utilizada para o desenvolvimento de aplicativos com componentes visuais contendo interface gráfica de usuário (GUI). Apresenta uma interface “amigável”, permitindo que qualquer pessoa crie aplicativos profissionais sem precisar ser um desenvolvedor de software profissional.

Um projeto criado a partir da ferramenta pode ser compartilhado com outros usuários do MATLAB via MATLAB Online e MATLAB Drive para que eles possam executar o aplicativo, colaborar em designs e ter permissão para editar os arquivos. Além disso, é permitido guardar o aplicativo em um único arquivo, criando um programa executável da interface com extensão (.mlappinstall). Os arquivos do instalador permitem que o usuário e outras pessoas instalem o aplicativo e acessem-no na galeria de aplicativos, sem necessidade de comprar a licença do software MATLAB® [14].

3 METODOLOGIA

O desenvolvimento da aplicação do sistema de calibração para o sensor de cor TCS230 foi realizado em duas etapas principais. Na primeira, definiu-se a estrutura física, a elaboração do código via IDE do Arduino e os primeiros testes de leitura do sensor. E a segunda etapa consistiu no desenvolvimento da interface gráfica de usuário (GUI), em código MATLAB com a opção *App Designer*, e testes para validar a aplicação.

3.1 O protótipo

A estrutura física onde o sensor de cor está inserido, é um projeto elaborado por Da Silva [7], cujo sistema trata do protótipo de uma máquina separadora de peças por cor. O protótipo, mostrado na Figura 2a e 2b, consiste em uma estrutura de madeira contendo um sensor de cor do tipo TCS230 e dois servos motores que estão

ligados ao microcontrolador Arduino, responsável pelo funcionamento do sistema, o que o torna viável para dimensão do projeto por tratar-se de um sistema em pequena escala [7].

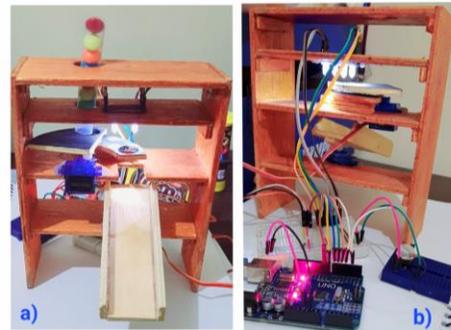


Figura 2: a) frente e b) verso do protótipo da máquina separadora de peças por cor.

A atuação do sistema para este trabalho consiste em quatro etapas principais.

- Etapa 1: Trata-se do início do processo, onde se encontra o coletor de peças, um tubo que recebe as peças coloridas.
- Etapa 2: Com um clique no botão o primeiro servomotor 1 recebe a peça e por meio de um segundo clique ao *push button* (botão de pressão), a peça é levada até o sensor de detecção de cores, TCS320.
- Etapa 3: Nesta etapa, o sensor de identificação de cores faz a leitura da escala de frequência enviada pelo objeto em análise. O sistema permanece nesse estado até o botão ser pressionado novamente.
- Etapa 4: O botão é pressionado uma terceira vez para dispensar a peça na rampa que leva ao descarte no recipiente. Nesse estágio, está o servomotor 2, que, nesta aplicação, não recebe comandos de movimentação. Seu uso é apenas para suporte da rampa, uma vez que o objetivo neste projeto não é a separação das peças em si.

Para a alimentação do protótipo, foram utilizados doces de chocolate coloridos (corpos de prova), que exibem cores uniformes e atendem aos requisitos do protótipo em tamanho e forma. Para que os confeitos pudessem ter um tempo de durabilidade maior e não tivessem suas características alteradas durante o uso, foi aplicado esmalte incolor para revestir cada peça.

Para este projeto, foram escolhidas cinco cores (vermelho, verde, azul, laranja e amarelo), e separados cinco confeitos de cada uma das cores, conforme mostra a Figura 3.

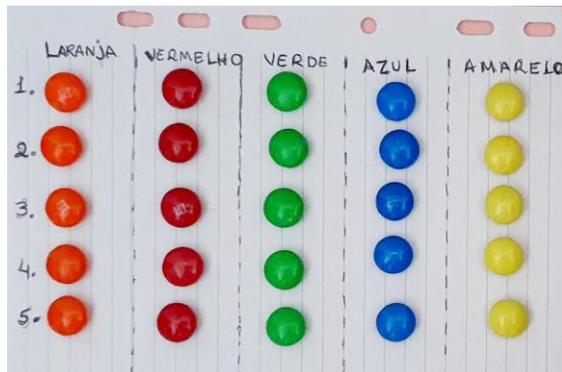


Figura 3: Corpos de prova organizados em sequência de testes.

3.2 Código e calibração do sensor

O código para o projeto, desenvolvido no Ambiente de Desenvolvimento Integrado do Arduino, desempenha um papel fundamental na comunicação entre a placa Arduino e o sensor. Ele permite a obtenção dos dados emitidos pelo sensor com o intuito de realizar a leitura desses valores no sistema de cores RGB, como destacado por Silva [4], devido ao fato de o sensor ler valores analógicos representativos do comprimento de onda emitido pela cor, é necessário realizar a calibração na programação. Essa calibração implica inserir, nas linhas de código, os intervalos de frequência emitidos pelo sensor correspondentes à cor a ser lida. Em outras palavras, é necessário realizar testes com corpos de prova, para determinar a faixa de valores que serão fornecidos ao programa.

No caso específico do sensor de cores TCS230, conforme mencionado por Oliveira et al. [6], a calibração foi realizada por meio de papéis impressos com tonalidades simuladas das cores necessárias utilizadas no projeto, e os espectros de cores foram lidos em RGB e tons de branco, possibilitando a separação das cores com base em parâmetros distintos por meio de condicionais no código. Esse processo não apenas facilita a leitura dos valores RGB medidos pelo sensor, mas também a determinação das faixas correspondentes a cada cor. Além disso, o software oferece a praticidade de modificar o código e testá-lo, possibilitando também a comunicação com outros softwares.

3.3 A interface gráfica

A interface gráfica do aplicativo foi desenvolvida com o uso da ferramenta *App Designer* do software MATLAB, que auxilia na aquisição de dados, plotagem de gráficos e criação de tabelas para o sistema de detecção de cores. A Figura 4 a seguir ilustra a interface gráfica desenvolvida para este projeto. Para o seu desenvolvimento, foram utilizadas ferramentas como *Lamps* (lâmpadas), *Edit text* (editor de texto), *Edit numeric* (editor numérico), botão *switch*, gráfico, tabela e *Label* (texto), presentes no ambiente *App Designer*.

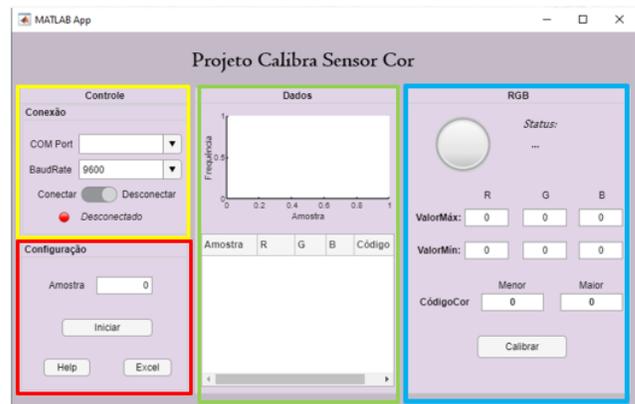


Figura 4: Janela da GUI desenvolvida para auxiliar na calibração do sensor cor

A interface conta com três painéis: o primeiro de controle, o segundo de recebimento dos dados e o terceiro para visualizar as informações de calibração do sensor. O painel de controle se divide em duas seções principais, na seção 1, marcada em amarelo, pode-se realizar a comunicação/conexão com o Arduino através de um botão *switch* (liga/desliga). A seção é acompanhada por um botão lâmpada (*Lamp*) que se mostrará verde para uma conexão bem-sucedida ou vermelho caso não haja conexão, bem como a presença de um texto que indica “conectado” e “desconectado”.

Na seção 2, circulado em vermelho e denominada configuração, há um campo de texto editável onde o usuário deve inserir a quantidade de amostras (leituras de frequência RGB) que deseja coletar e o botão de ‘Iniciar’, para que a aquisição de dados seja realizada. Além disso, a seção conta com o botão ‘Help’ que direciona a um arquivo com instruções para auxiliar os usuários no funcionamento do aplicativo e o botão ‘Excel’ para enviar os dados recebidos para uma planilha do Microsoft Excel caso seja necessário análises posteriores.

No segundo painel, marcado em verde, o usuário visualizará os dados recebidos através de um gráfico e de uma tabela que reproduz a leitura dos dados enviados pelo Arduino e a quantidade de amostra solicitada pelo usuário.

E no terceiro painel, destacado em azul, encontram-se os campos “valorMax” e “valorMin”, que apresentam as frequências máximas e mínimas das componentes R, G e B, obtidas ao final do monitoramento das amostras solicitadas. O campo “códigoCor” exibe o código que foi atribuído a cor (para cada cor neste projeto foi criado um código numérico de 1000 a 1500 no programa Arduino), proporcionando ao usuário uma verificação simples da eficácia da calibração. Se os valores máximos e mínimos do código para uma determinada cor forem idênticos, considera-se que o processo de calibração foi bem-sucedido; caso contrário, uma nova coleta de dados é necessária. Adicionalmente, neste painel, encontra-se o botão “Calibrar”, cuja função é iniciar o processo de calibração. Uma lâmpada indicadora mostra visualmente a cor detectada, enquanto um texto exibe o nome

correspondente a esta cor, proporcionando uma representação clara e intuitiva das informações sobre a calibração em execução.

4 RESULTADOS

Nesta seção serão apresentados e discutidos os resultados obtidos. O foco do trabalho foi dado para a implementação de um sistema que tornasse a calibração do sensor de cor mais cômoda e rápida, por meio do desenvolvimento de um aplicativo com o *App Designer*.

Para a verificação da confiabilidade do sistema foram realizados três tipos de testes para calibrar o sensor. No primeiro, a calibração foi realizada de forma empírica, onde os dados foram coletados manualmente por meio do monitor serial. O segundo e o terceiro foram realizados utilizando o Sistema ‘Projeto Calibra Sensor Cor’, um com ambiente iluminado e outro com ambiente sem iluminação, respectivamente. Todos os testes foram realizados com o mesmo sensor, e o principal objetivo foi observar o índice de defeitos detectados ou não detectados, de acordo os intervalos criados a partir dos testes.

Os corpos de prova de cada cor foram inseridos individualmente sobre o sensor. Para cada caso, as frequências detectadas e não detectadas foram registradas e organizadas em tabelas. Depois de recolher todas as informações recebidas, os resultados foram comparados e analisados para tirar conclusões sobre a eficiência do aplicativo.

4.1 Teste I – Coleta de dados em formato manual

Inicialmente, com auxílio do monitor serial do Arduino e um tempo de amostragem de 10 segundos, foram retiradas 10 (dez) leituras das frequências RGB emitidas pelo sensor para cada uma das peças. Os valores obtidos foram adicionados em planilhas do Microsoft Excel, e dispostos em tabelas. Através das funções "Média", "Mínimo" e "Máximo", foi calculado a média, o máximo e o mínimo dos valores obtidos em R, G e B de cada um dos corpos de prova. A partir desta análise, foi criado o intervalo da frequência do sinal recebido pelo sensor para inserir nas linhas de código que permitem a calibração do sensor. Os resultados podem ser visualizados na Tabela 1.

Tabela 1: Intervalos das frequências coletadas com o Teste I.

Cores	Filtro		
	R	G	B
Vermelho	170-135	285-240	215-170
Verde	200-150	200-150	200-150
Azul	235-189	215-160	137-110
Laranja	125-90	240-210	205-180
Amarelo	115-90	140-110	155-135

Com os valores da Tabela 1 adicionados às linhas de Código, foi feita a análise de detecção para as dez amostras de frequências coletadas. A Tabela 2 mostra os resultados obtidos por meio da

leitura realizada para todas os corpos de prova. A nomenclatura “D” é para Detectados e “ND” é para Não Detectados. O total de amostra refere-se às leituras de frequência RGB obtidas do sensor.

Tabela 2: Resultados de peças detectadas e não detectadas no Teste I.

Quantidade de Peças	Total de Amostra	Vermelho		Verde		Azul		Laranja		Amarelo	
		D	ND	D	ND	D	ND	D	ND	D	ND
1	10	7	3	10	0	9	1	9	1	7	3
2	10	8	2	8	2	7	3	4	6	10	0
3	10	8	2	10	0	1	9	10	0	9	1
4	10	10	0	7	3	9	1	1	9	9	1
5	10	8	2	8	2	10	0	8	2	9	1

4.2 Teste II – Coleta de dados com uso do aplicativo: ambiente iluminado

Os experimentos foram repetidos, agora utilizando o sistema ‘Projeto Calibra Sensor Cor’, com base nos mesmos critérios mencionados no Teste I em relação ao tempo e à quantidade de amostras. Os resultados apontaram um novo intervalo de frequências, conforme Tabela 3.

Tabela 3: Intervalos de frequências obtidas no Teste II.

Cores	Filtro		
	R	G	B
Vermelho	180-130	300-240	280-170
Verde	280-150	280-150	280-150
Azul	255-189	220-190	150-120
Laranja	128-90	245-210	215-165
Amarelo	125-90	150-102	160-130

Para validar o sistema, foi efetuada uma nova análise de detecção com os intervalos obtidos no Teste II. Com base nas leituras obtidas, houve uma melhora nos resultados em relação ao teste anterior com a implementação das novas frequências. Todas as peças de cor vermelha foram 'detectadas', enquanto para as cores verde e azul, ocorreu a detecção de nove peças. Além disso, os 'não detectados' foram reduzidos para as cores amarela e laranja, como mostra a Tabela 4, resultado da análise feita para todos os corpos de prova.

Tabela 4: Resultados de peças detectadas e não detectadas no Teste II.

Qnt.Peças	Amostra	Vermelho		Verde		Azul		Laranja		Amarelo	
		D	ND	D	ND	D	ND	D	ND	D	ND
1	10	10	0	10	0	10	0	10	0	10	0
2	10	10	0	10	0	9	1	10	0	9	1
3	10	10	0	9	1	10	0	9	1	10	0
4	10	10	0	10	0	10	0	9	1	9	1
5	10	10	0	10	0	10	0	10	0	10	0

4.3 Teste III – Coleta de dados com o aplicativo: ambiente escuro

O Teste III foi realizado em um ambiente sem iluminação (sem luz externa), seguindo os mesmos critérios do Teste II. Novamente, foi necessário criar um intervalo de frequências para que a detecção dos corpos de prova fosse satisfatória neste novo cenário. Através da análise de cada tabela os intervalos foram alterados, conforme a Tabela 5, para cada cor, exceto a cor verde que teve as peças detectadas com as mesmas informações do Teste II.

Tabela 5: Resultados de frequências coletadas no Teste III.

Cores	Filtro		
	R	G	B
Vermelho	200-130	400-240	380-170
Verde	280-150	280-150	280-150
Azul	355-189	220-190	150-120
Laranja	128-90	245-210	215-165
Amarelo	129-90	245-200	215-165

Em um ambiente sem luminosidade externa, algumas peças de cor laranja apresentaram características semelhantes às peças de cor vermelha. Isso torna a calibração dos corpos de prova bastante desafiadora, pois, para evitar a detecção de peças vermelhas em ambientes com ausência de luz externa, os intervalos de frequências devem ser muito específicos, o que pode resultar na não detecção de muitas peças.

De modo semelhante, ocorre com peças de cor amarela, onde algumas peças não foram detectadas e outras emitem frequências que são detectadas como peças verdes.

A Tabela 6 mostra os resultados obtidos através da leitura realizada para todos os corpos de prova, com o novo intervalo de frequência, para determinar a resposta de peças detectadas e não detectadas. As peças laranjas e amarelas detectadas com outras colorações foram consideradas como não detectadas.

Tabela 6: Resultados de detecção das peças para Teste III

Cor	Vermelho		Verde		Azul		Laranja		Amarelo	
	D	ND	D	ND	D	ND	D	ND	D	ND
1	10	0	10	0	10	0	9	1	9	1
2	10	0	10	0	10	0	10	0	10	0
3	10	0	10	0	10	0	10	0	9	1
4	10	0	10	0	10	0	7	3	9	1
5	10	0	10	0	10	0	8	2	8	2

Diante das análises realizadas, percebe-se que as frequências emitidas pelo sensor em ambiente com menos luminosidade têm os valores mais altos, enquanto em ambientes com luminosidade externa os valores são mais baixos. A Tabela 7 apresenta uma média percentual da variação de redução desses valores.

Tabela 7: Média percentual da variação de redução dos valores de frequências emitidas pelo sensor.

	Média Percentual				
	Vermelho	Verde	Azul	Laranja	Amarelo
R	10%	6%	7%	4%	3%
G	9%	5%	3%	7%	3%
B	7%	4%	1%	4%	2%

4.4 Desempenho do aplicativo

A calibração de forma manual se mostra satisfatória, uma vez que o sistema não falha em todas as cores. No entanto, de acordo com as análises, percebe-se que, quando conduzida manualmente por meio de análise humana, essa abordagem pode introduzir uma taxa mais elevada de erros na medição, além de requerer mais tempo e dedicação em comparação com a inspeção realizada pelo sistema desenvolvido (Projeto Calibra Sensor Cor). O aplicativo se destaca pela capacidade de trabalhar com um número preciso de amostras e de forma consecutiva.

Quando os dados são gerados, o gráfico mostra o comportamento das frequências emitidas para a cor que está sob o sensor. O aplicativo possibilita a visualização gráfica das frequências máximas e mínimas, permitindo compreender o comportamento do sensor. Dessa forma, é possível identificar facilmente quais as frequências em R, G e B que se alteram ao longo das medições requeridas.

No terceiro painel os campos de “códigoCor”, assim como a *Lamp* e o “Status” indicam os dados que permitem calibrar o sensor de acordo com os intervalos captados para a peça em análise. Isso significa que o usuário deve observar os valores máximos e mínimos e fazer as alterações necessárias nas linhas de código no IDE do Arduino.

A Figura 5 mostra a tela do aplicativo em testes realizados para cor vermelha, em que o cenário mostra que os intervalos de frequências adicionadas nas linhas de código do Arduino calibram o sensor.

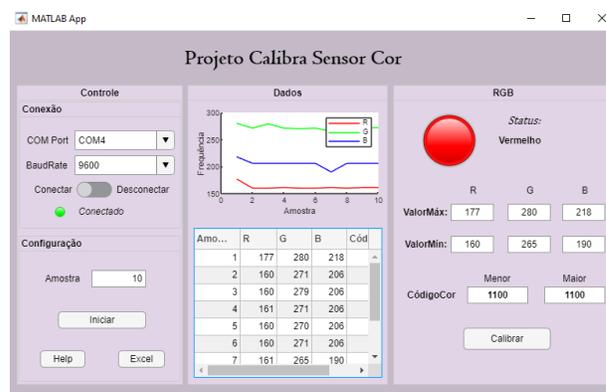


Figura 5: Resultado da calibração para cor vermelha.

É importante observar que o sistema não utiliza conceitos de aprendizado de máquina, dependendo de valores de treinamento, o que pode resultar em erros na identificação das cores e limitar a eficácia em análises comparativas. Além disso, a codificação das cores no aplicativo ainda apresenta limitações, pois está sujeita a erros ao ser programada para os valores em máximo ou mínimo.

5 CONCLUSÃO

A aplicação desenvolvida no âmbito deste estudo revelou-se uma ferramenta eficiente para o propósito pretendido, permitindo a aquisição de dados de forma automatizada, eliminando a necessidade de seleção manual de amostras e fornecendo valores máximos e mínimos essenciais para as escalas de frequência. Os testes realizados demonstraram a capacidade do aplicativo de identificar intervalos de cores adequados para até três peças da mesma cor, simplificando a definição dos intervalos para análise de cores, de acordo com as informações do ambiente.

Destaca-se que o aplicativo não se limita apenas a verificar valores da variação de luminosidade do ambiente, mas inserir uma metodologia que auxilia na definição dos intervalos, em análise de dados, e permitir que haja comodidade, diminuindo o tempo gasto no processo de calibração do sensor.

Em resumo, o "Projeto Calibra Sensor Cor" apresenta-se como uma ferramenta valiosa para a calibração de sensores de cor do tipo TCS230 ou similares. Com recursos de orientação, como um manual de instruções e a capacidade de exportar dados para o Microsoft Excel, esta plataforma é versátil e pode ser utilizada em diversas aplicações que requerem análise precisa de cores. Sua eficiência e usabilidade tornam-no uma contribuição significativa para o campo do sensoriamento de cores e análise de dados.

Como recomendação de trabalhos futuros, propõe-se apresentar um modelo computacional que seja capaz de criar soluções de programação tanto para o Arduino quanto no MATLAB, tornando o processo de calibração totalmente automático.

REFERÊNCIAS

- [1] Pedro Henrique Santos de Medeiros. Controle de uma maquete de ponte rolante com interface gráfica no MATLAB utilizando o microcontrolador ATMEGA 2560 embarcado em uma plataforma Arduino. Trabalho de Conclusão de Curso, Universidade Federal do Espírito Santo (Ufes), Vitória, Es, jul 2017.
- [2] Poltak SIHOMBING, Faddly Tommy, Sajadin Sembiring and Nogar Silitonga. The citrus fruit sorting device automatically based on color method by using tcs320 color sensor and arduino uno microcontroller. In: *Journal of Physics: Conference Series* 1235:012064. IOP Publishing, 2019. p. 012064. doi: 10.1088/1742-6596/1235/1/012064.
- [3] Andrada Cassio Maia. Elaboração de protótipo seletor de grãos de café, baseado em sensor de cores rgb para se determinar o início da colheita. Trabalho de conclusão de curso. Faculdades Doctum de Caratinga, Caratinga, 2019.
- [4] Fabrício Dias da Silva. Desenvolvimento De Uma Plataforma Para Caracterização De Cores E Índice De Maturação Em Goiaba (Psidium Guajava L.) Utilizando Um Microcontrolador Programável Arduino. Trabalho de Conclusão de Curso, Universidade Federal de Campina Grande (UFCG), Pombal, Pb, 2021.
- [5] Rodrigo Alencar de Souza. Desenvolvimento de protótipo de sistema para monitoramento contínuo da urina humana. Trabalho de conclusão de curso. Universidade do Estado do Amazonas, Manaus, 2022.
- [6] Denimar Souza de Oliveira, Leo Masaharu Amaya Kitagawa, Victor Arthur Ravani. Sistema de identificação de cédulas monetárias. Trabalho de conclusão de curso. FACULDADE DE TECNOLOGIA DE SÃO PAULO – FATEC-SP, São Paulo, 2020.
- [7] Adonias Wanderson Duarte Da Silva. Automação De Um Sistema De Seleção De Peças Por Cores Com Arduino E Supervisório Scada. Trabalho de conclusão de curso. Universidade Federal do Oeste do Pará (Ufopa), Santarém, PA, 2019.
- [8] Daniel Thomazin; Pedro Urbano Braga de Albuquerque. Sensores industriais: fundamentos e aplicações. Saraiva Educação SA, 2013.
- [9] Jonas Greschuk; Jefferson Lucas Pauli. Sistema de reconhecimento de objetos por cor integrado a uma mesa pneumática de coordenadas no plano cartesiano (X,Y) - Joinville Instituto Federal de Santa Catarina, 2013. 54f. Trabalho de Conclusão de Curso - Instituto Federal de Santa Catarina, 2015.
- [10] Sensor de cores RGB TCS3200. Disponível em: <https://orangemaker.com.br/produto/sensor-de-cores-rgb-tcs3200/>. Accessed: 2023-09-23
- [11] TAOS LUMENOLOGY. Datasheet: Tcs230 Programmable Color Lighttofrequency Converter. 2008. URL. <https://pdf1.alldatasheet.com/datasheet-pdf/view/202765/TAOS/TCS230.html>.
- [12] Mariana Emer de Queiroz. Elaboração E Implementação De Um Protótipo De Esteira Seletora Por Cor. Trabalho de Conclusão de Curso, Universidade Tecnológica Federal do paraná, Campo Mourão, 2017.
- [13] Michael McROBERTS, Arduino Básico (Novatec Editora Ltda., São Paulo, 2011).
- [14] MATHWORKS. MATLAB App Designer. 2022. <https://www.mathworks.com/products/matlab/app-designer.html>. Accessed: 2023-09-20