

## **Desenvolvimento de uma plataforma para identificação de deformidades de produtos em linhas de produção empregando técnicas de *machine learning***

### **Development of a platform to identify product deformities in production lines using *machine learning* techniques**

DOI:10.34117/bjdv8n5-259

Recebimento dos originais: 21/03/2022

Aceitação para publicação: 29/04/2022

#### **Marcony Montini de Oliveira Lima**

Graduado em Engenharia Mecatrônica e Graduando em Engenharia da Computação

Instituição: Universidade do Estado de Minas Gerais (UEMG)

Endereço: Av. Paraná, 3001, Bairro Jardim Belvedere I, CEP: 35501-170

Divinópolis, MG

E-mail: marconymol@hotmail.com

#### **Thiago Magela Rodrigues Dias**

Doutor em Modelagem Matemática e Computacional

Instituição: Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG)

Endereço: R. Álvares de Azevedo, 400 – Bairro Bela Vista, CEP: 35503-822

Divinópolis – MG

E-mail: thiago.cefetmg@gmail.com

#### **RESUMO**

A proposta do presente projeto foi o desenvolvimento de uma plataforma para análise de deformidades, contemplando as áreas da Computação e Controle. Com o desenvolvimento de algoritmos, empregando a linguagem de programação Python, teve-se por objetivo apresentar uma solução para problemas em inspeção de produtos em linhas de produção, uma vez que, em alguns casos, o trabalho manual é complexo, demorado ou repetitivo. A identificação das principais deformações apresentadas por um produto em sua linha de produção foi essencial para o treinamento da plataforma, efetuado através do processamento e análise de imagens, utilizando métodos de aprendizado de máquina. Foram definidos como objetos de análise ampolas/refis de insulina para canetas de aplicação. A razão desta escolha se deu por este material ser de fácil manuseio, visando verificar a funcionalidade da plataforma para identificação de variações de cores e volume do fluido, além da sua alta aplicação na área farmacêutica, abordando multidisciplinaridade ao projeto. Por fim, é importante ressaltar que foram obtidos resultados satisfatórios com valores interessantes de acurácia, porém, não foram realizados testes de aplicação em tempo real.

**Palavras-chave:** *Machine learning*, redes neurais artificiais, análise de deformidade.

#### **ABSTRACT**

The purpose of this project was the development of a platform for the analysis of deformities, covering the areas of Computing and Control. With the development of algorithms, using the Python programming language, the objective was to present a solution to problems in inspection of products in production lines, since, in some cases, manual work is complex, time-consuming or repetitive. The identification of the main

deformations presented by a product in its production line was essential for the training of the platform, carried out through the processing and analysis of images, using machine learning methods. Insulin ampoules/refills for application pens were defined as objects of analysis. The reason for this choice was due to the fact that this material is easy to handle, aiming to verify the functionality of the platform to identify variations in colors and fluid volume, in addition to its high application in the pharmaceutical area, approaching multidisciplinary to the project. Finally, it is important to note that satisfactory results were obtained with interesting accuracy values, however, real-time application tests were not performed.

**Keywords:** *Machine learning*, artificial neural networks, deformity analysis.

## 1 INTRODUÇÃO

O processo de automação tem crescido cada vez mais na área industrial, uma vez que proporciona menor tempo para se realizar uma tarefa, maior rendimento e melhor qualidade, quando empregado de forma correta, se comparado ao serviço braçal (Araújo Júnior, et al. 2003). Em uma linha de produção, automatizada ou não, existe a possibilidade do produto apresentar deformidades ou irregularidades, podendo estas terem sido originadas no processo de transporte ou transformação da matéria-prima empregada, antes mesmo de chegar na linha de montagem. A verificação destas inconformidades é, em alguns casos, realizada de forma visual, onde um funcionário acompanha o trajeto dos produtos.

Diversos sistemas de automação e controle têm sido desenvolvidos a fim de otimizar o processo em uma linha de produção. O alto desempenho e processamento dos atuais sistemas eletroeletrônicos possibilitam uma excelente performance na solução de problemas reais através de respostas rápidas. Quando um produto e/ou serviço não atendem às especificações de qualidade do cliente, existe uma perda no processo de fabricação. Estabelecer a diferença entre verificar para “prevenir” produtos defeituosos e verificar para “localizar” defeitos é essencial para se reduzir estas perdas (Abreu, 2002).

Existe uma ligação íntima entre a questão de qualidade e a análise das perdas para sua eliminação. Reduzindo estas perdas, é possível gerar recursos que poderão ser aplicados em formas de alavancar o sistema de qualidade, trazendo benefícios que, futuramente, serão capazes de suplantarem os gastos iniciais (Robles, 1994).

Logo, este trabalho almejou o desenvolvimento de uma plataforma que atuasse na detecção de deformidades superficiais em peças ou produtos em uma linha de produção, através da análise realizada a partir da imagem obtida por câmeras. A

classificação das imagens é efetuada a partir de técnicas de aprendizado de máquina, ou *machine learning*. Ao se identificar a presença de irregularidades no produto, o mesmo pode ser retirado da linha de montagem e separado dos demais através da aplicação de um atuador simples, por exemplo.

Com a plataforma proposta, espera-se apresentar uma opção de ferramenta para manter a uniformidade na linha de produção. Supondo-se que seja implementado um canal de comunicação com um atuador que possa retirar o produto danificado da linha no momento de sua identificação, sem gerar atrasos durante a análise dos produtos, pode ser possível também exprimir os tipos de deformidades mais comuns encontrados e resultar em uma concebível solução para suprimi-las. Não foi o objetivo deste trabalho aplicar a conexão a um atuador, mas é um possível cenário de aplicabilidade à plataforma. Tais processos, conseqüentemente, gerariam uma redução de custos em materiais que seriam descartados.

## 2 METODOLOGIA

Este projeto tem como objetivo apresentar uma plataforma para inspeção de objetos ou peças em uma linha de produção. A aquisição de dados a serem analisados é realizada através de câmeras que, como componentes de sistema de aquisição, possuem uma matriz de sensores sensíveis a luz (Stivanello, 2004). Com a aplicação de luz sobre o objeto de análise, a sua reflexão incide sobre a câmera, gerando um valor de intensidade em cada coordenada da matriz, formando uma imagem digital. Para tal, foi utilizada a câmera e componente de iluminação de um *smartphone* com resolução de 4160x3120 pixels.

A técnica de iluminação a ser empregada varia de acordo com o resultado a ser buscado, ressaltando diferentes características do produto. Utilizando um anel luminoso em torno da câmera, por exemplo, é possível trabalhar com reconhecimento de caracteres. Outro exemplo é a iluminação por trás do produto, que proporciona uma imagem destacando sua silhueta, útil para análise de suas bordas.

Em se tratando da aplicação de *machine learning*, o objetivo é encontrar e classificar padrões, além de generalizar as informações, através da linguagem de programação Python. Neste quesito, arbitrariamente, optou-se por utilizar redes neurais artificiais e máquinas de vetores de suporte, com o objetivo de comparar e identificar qual método apresentaria melhor performance.

Para o desenvolvimento da rede neural artificial (RNA), primeiramente, é realizada a sua definição, considerando seu tamanho, o tipo de problema a ser resolvido e o tipo de aprendizado do algoritmo. Em seguida, inicia-se a etapa de treinamento supervisionado, em que são apresentados pares de situações onde, para cada entrada, a saída desejada também seja mostrada. Se a saída obtida se difere da desejada, é realizado um “aprendizado” da rede. Se a saída obtida for igual à desejada, é apresentado um novo par. Este procedimento é repetido diversas vezes, aumentando a probabilidade de acerto do algoritmo. Por fim, é realizada sua implementação.

Se tratando das máquinas de vetores de suporte (*support vector machines* – SVM), esta técnica trata de classificar cada uma das amostras apresentadas na etapa de treinamento através de separação linear. Com base nesta separação, é realizada uma comparação entre as amostras de treinamento e de teste, gerando uma classificação. Como aplicação para a implementação da plataforma, foi definida a classificação de um objeto de análise específico: ampolas de refil de canetas para aplicação de insulina do tipo humana regular.

Como prova de conceito, foi utilizado um refil vazio do medicamento NovoRapid®. Esta é uma insulina moderna (insulina análoga) de ação ultra-rápida. Insulinas modernas são versões aprimoradas da insulina humana. Este medicamento é usado para tratar diabetes mellitus em adultos, adolescentes e crianças (a partir de 2 anos de idade) (NovoRapid® FlexPen®, 2017). Diabetes mellitus é uma doença na qual seu organismo não produz insulina suficiente para controlar o nível de açúcar no sangue. Uma das formas de aplicação é através de uma caneta, com uma agulha na ponta, em que o medicamento é adicionado à caneta através de um refil. Conforme é possível observar na Figura 1, para treinamento e testes, utilizou-se imagens do refil contendo diferentes soluções, colorações e volumes.

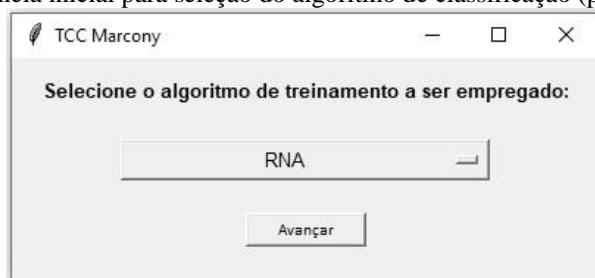
Figura 1 - Exemplos de amostras utilizadas para classificação (próprio autor).



### 3 DESENVOLVIMENTO

Primeiramente, em relação à interação com o usuário, optou-se pelo desenvolvimento de uma interface gráfica. Para tal, foi selecionado o Tkinter, que nada mais é do que uma biblioteca da linguagem Python específica para desenvolvimento de GUIs (*Graphic User Interface* – Interface Gráfica com o Usuário) (Ferg, 2009). Esta aplicação foi necessária para auxiliar o usuário na definição de parâmetros, uma vez que a alteração direta dos parâmetros em linha de código pode ser confusa e carecer de certo conhecimento básico de programação. A Figura 2 apresenta a janela inicial da Interface, onde é realizada a definição do algoritmo de classificação a ser empregado. Após esta janela inicial, é apresentada uma janela onde são definidos os parâmetros principais para treinamento e o método de tratamento de dados. Nesta segunda janela também é realizada a seleção dos arquivos em formato .csv com as respostas esperadas, utilizadas para treinamento, que correspondem a uma matriz coluna, onde cada linha corresponde à resposta esperada de uma figura.

Figura 2 - Janela inicial para seleção do algoritmo de classificação (próprio autor).



Ao definir os parâmetros adequadamente, autenticá-los e clicar no botão “Treinar”, o treinamento é realizado. É exibido o valor de acurácia do algoritmo em relação aos dados do próprio treinamento e são apresentadas as opções de teste para uma imagem específica (retornando sua previsão) ou para várias (que calcula a acurácia do algoritmo para um conjunto de respostas pré-definidas).

Em relação ao processamento e tratamento de dados, notou-se que a quantidade exuberante de variáveis de entrada (20736, para imagens em escala 192x108 pixels e 129600, para 480x270 pixels, por exemplo) era um fator decisivo, tanto para o desempenho do código em questão de tempo de execução, quanto para questões de acurácia. Um número maior de entradas exige uma quantidade elevada de neurônios e, conseqüentemente, mais pesos a serem atualizados a cada iteração. Da mesma forma,

existe chances de, quanto maior o número de entradas normalizadas, menor ser a variância entre elas, dificultando a previsão correta do algoritmo.

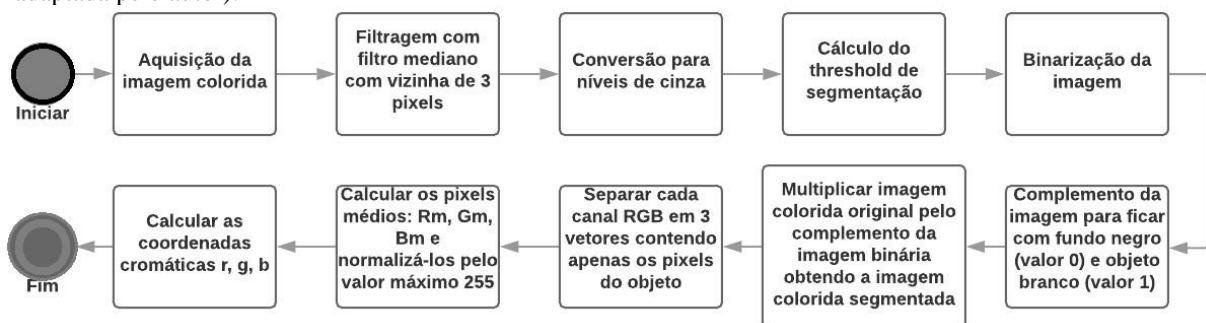
$$r = \frac{\sum_{i=1}^N R_i}{\sum_{i=1}^N R_i + \sum_{i=1}^N G_i + \sum_{i=1}^N B_i} \quad (1)$$

$$R_m = \frac{\left(\frac{\sum_{i=1}^N R_i}{N}\right)}{255} \quad (2)$$

Para a redução significativa do número de variáveis de entrada, optou-se por utilizar três métodos diferentes, objetivando-se identificar qual resultaria em um melhor resultado após aplicação da classificação dos dados. Foram eles o PCA (*Principal Component Analysis*), o SVD (*Singular Value Decomposition*) e o método de tratamento por segmentação apresentado no fluxograma da Figura 3.

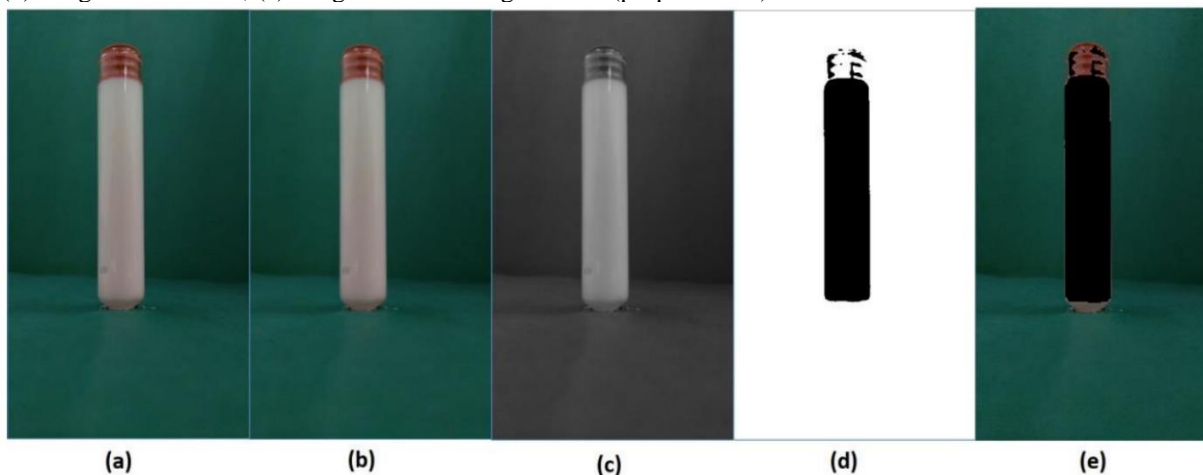
Para este terceiro método, as coordenadas cromáticas foram calculadas através da equação (1) e os pixels médios através da equação (2) (Louro, et al. 2006), onde, neste caso, estão sendo calculados os valores para o canal vermelho, representado pela letra R. Para a coordenada cromática referente ao canal vermelho, é realizada a somatória de todos os pixels e o resultado é dividido pela soma das somatórias de cada canal. Para os pixels médios, é realizada a somatória de todos os pixels do canal, onde o resultado é dividido pelo número total N de pixels e normalizado, sendo dividido por 255.

Figura 3- Fluxograma do tratamento de imagem para extração de características (Louro, et al.2006, adaptada pelo autor).



A Figura 4 apresenta um exemplo de tratamento de uma das imagens abordadas para classificação do refil de canetas para aplicação de insulina, aplicando o método de segmentação.

Figura 4 - Exemplo de tratamento de imagem, representando: (a) a imagem original colorida; (b) imagem após filtragem com filtro mediano com vizinhança de 3 pixels; (c) imagem convertida para níveis de cinza; (d) imagem binarizada; (e) imagem colorida segmentada (próprio autor).

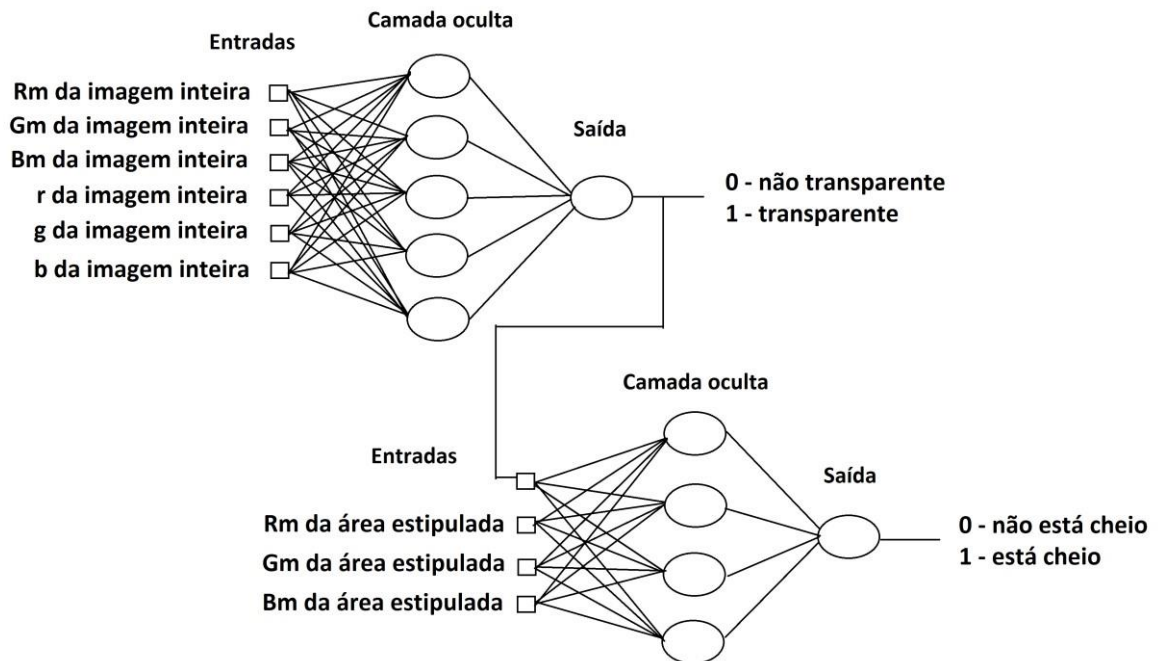


Desta forma, foram utilizadas como variáveis de entrada do algoritmo de classificação os pixels médios e as coordenadas cromáticas, reduzindo o número de entradas para apenas 6. Assim, foram observadas duas características para classificação: a coloração do fluido contido em seu interior e o seu volume. Através da coloração é possível notar se o produto está vencido, uma vez que a coloração normal para a insulina regular é transparente e, qualquer coisa diferente disso, não é aceitável. Se o fluido apresentar cor turva ou esbranquiçada, está vencido (Humulin®, 2018), e, se apresentar qualquer outra variação de cor, especula-se que pode ter ocorrido algum erro nos processos químicos. Em relação à quantidade, não é desejável que o produto, ao ser comprado, apresente menor volume do que o especificado em sua embalagem.

Se tratando da aplicação de RNAs, visando classificar o produto através destas duas características, foram empregadas duas redes neurais sequencias: a primeira responsável por tratar da verificação da coloração e a segunda da análise da altura. A primeira rede neural apresenta 6 entradas, uma camada oculta de número de neurônios definido pelo usuário e uma camada de saída com um neurônio. Conforme é possível observar na Figura 5, a saída da primeira classificação é dada como entrada na segunda rede neural, que possui também como entrada o valor de altura do fluido.

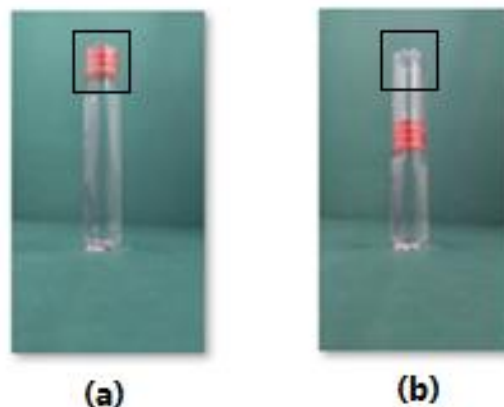
Da mesma forma, para o SVM, também foram aplicados dois treinamentos e classificações sequenciais.

Figura 5 - Rede neural implementada para classificação dos dados após tratamento especificado (Louro, et al. 2006, adaptada pelo autor).



Para verificação da altura, por meio da Figura 6, é possível observar que foi utilizada como referência a posição da borracha utilizada como tampa quando o refil está cheio, já que a sua posição acompanha o volume do fluido. Quando o volume interno da ampola diminui, a borracha acompanha o movimento de descida, logo, quando o fluido estiver abaixo do nível considerado como ideal, a borracha não estará dentro da área estipulada. Assim, é feito o cálculo da média dos pixels dentro desta área para verificação da altura e o resultado é dado como entrada na segunda rede neural.

Figura 6 – (a) Ampola cheia, onde a borracha está posicionada dentro da posição de referência devido ao volume adequado de fluido em seu interior. (b) Ampola com fluido faltante, onde a borracha nitidamente está fora da posição que representa uma ampola cheia. (Fonte: próprio autor).





#### 4 RESULTADOS E DISCUSSÕES

Um mesmo objeto foi fotografado diversas vezes, de forma a possuir variações a cada imagem adquirida, com coloração e volume diferentes. A insulina de cor aceitável é incolor (rótulo 1), sendo qualquer outra entrada considerada inadmissível. Da mesma forma, apenas refis com volume em seu total seriam tolerados. Três bancos de dados foram utilizados para treinamento, sendo que, em todos, a câmera foi posicionada à 8,5cm de distância do objeto. Outro banco de dados foi utilizado para a fase de teste, com imagens diferentes das apresentadas na fase de treinamento. A Tabela 1 apresenta a relação de imagens em cada banco de dados.

Tabela 1 - Relação de amostras nos bancos de dados (próprio autor).

Treinamento 1		Treinamento 2		Treinamento 3		Teste 1	
Transparente	6	Transparente	58	Transparente	26	Transparente	12
Não transparente	34	Não transparente	22	Não transparente	94	Não transparente	52
Cheio	6	Cheio	6	Cheio	6	Cheio	6
Não cheio	34	Não cheio	74	Não cheio	114	Não cheio	58
<b>Total</b>	<b>40</b>	<b>Total</b>	<b>80</b>	<b>Total</b>	<b>120</b>	<b>Total</b>	<b>64</b>

Para cada banco de dados foi realizada a classificação através dos algoritmos de RNA e SVM, sendo tratados por PCA, SVD ou segmentação da imagem. Também foram observados os tempos de execução do algoritmo para as etapas de treinamento e teste para diferentes escalas, conferindo assim sua viabilidade computacional. Utilizando, ao banco de dados de treinamento 1 e ao de testes 1, o algoritmo RNA com processamento para redução de dados pelo método SVD com escala de 1:10, foram necessários 21 segundos para a fase de treinamento e 33 segundos para a fase de teste. Para a escala 1:5, 1 minuto e 2 segundos para o treinamento e 1 minuto e 40 segundos para o teste. Em escala 1:2, 6 minutos e 19 segundos para o treinamento e 37 minutos e 52 segundos para o teste.

Notou-se ser inviável a aplicação da escala 1:1, uma vez que o tempo para treinamento e teste era muito elevado e a acurácia do algoritmo apresentava variações mínimas em relação à escala para os primeiros três testes, com valores em torno de 76% e 81%. Desta forma, foi definido aplicar apenas a escala 1:10.

Todos os parâmetros aplicados ao treinamento – como taxa de aprendizagem, alfa, etc – foram definidos de forma empírica, verificando quais resultavam em melhor acuraria. Assim, verifica-se a dinâmica de cada algoritmo e tratamento de forma ponderada.

Para o primeiro caso, conforme Tabela 2, os resultados foram muito bons para testes realizados com os dados utilizados no próprio treinamento da RNA, porém, não foram satisfatórios para dados diferentes (dados do teste). Apenas o banco de dados com 120 amostras apresentou resultado aceitável (acima de 80% de acurácia). Tal resultado se deve ao maior número de imagens na fase de treinamento.

Tabela 2 - Relação de acurácia (%) no treinamento por RNA e tratamento SVD para as características de coloração e volume (próprio autor).

RNA - Tratamento por SVD						
Nº de amostras	Acurácia ao próprio treinamento			Acurácia ao teste		
	Coloração (%)	Volume (%)	Tempo	Coloração (%)	Volume (%)	Tempo
40	100	100	21s	73,44	76,56	33s
80	100	93,75	45s	78,12	89,06	37s
120	100	95,83	1min20s	82,81	87,5	37s

Em seguida, pelo tratamento PCA, foi realizado o treinamento RNA, conforme Tabela 3, em todos os bancos de dados. Para este caso também foram apresentados resultados satisfatórios diretamente proporcionais à quantidade de amostras, quando se tratando aos dados de teste.

Tabela 3 - Relação de acurácia (%) no treinamento por RNA e tratamento PCA para as características de coloração e volume (próprio autor).

RNA – Tratamento por PCA						
Nº de amostras	Acurácia ao próprio treinamento			Acurácia ao teste		
	Coloração (%)	Volume (%)	Tempo	Coloração (%)	Volume (%)	Tempo
40	100	100	27s	75	84,38	36s
80	100	93,75	44s	84,38	87,5	34s
120	100	95,83	1min06s	92,19	87,5	36s

O resultado da aplicação do tratamento de segmentação da imagem colorida aos bancos de dados para treinamento pelo algoritmo RNA pode ser observado na Tabela 4. Apesar deste método apresentar resultados mais baixos em relação aos dados do próprio treinamento, ele apresentou resultados mais satisfatórios ao teste, mantendo certa estabilidade aos valores, independentemente da quantidade de amostras no banco de dados. Isto se deve, provavelmente, porque o algoritmo não apresentou *overfitting*, ou seja, não houve uma adaptação ao banco de dados apresentado no treinamento, não havendo a perda da capacidade de generalização. Com base nestes resultados, é possível supor que uma maior variedade de dados poderia ocasionar em uma melhora da acurácia.

Tabela 4 - Relação de acurácia (%) no treinamento por RNA e tratamento de segmentação para as características de coloração e volume (próprio autor).

RNA – Tratamento por Segmentação						
Nº de amostras	Acurácia ao próprio treinamento			Acurácia ao teste		
	Coloração (%)	Volume (%)	Tempo	Coloração (%)	Volume (%)	Tempo
40	85	100	1min10s	73,44	85,94	1min52s
80	82,5	93,75	2min23s	73,44	87,5	1min52s
120	80	95,83	3min36s	81,25	89,06	1min52s

O próximo passo foi realizar os devidos treinamentos e testes utilizando o algoritmo SVM. A Tabela 5 apresenta os resultados para o tratamento SVD, onde nota-se certa estabilidade nos valores de acurácia. Este resultados serão abordados posteriormente.

Tabela 5 - Relação de acurácia (%) no treinamento por SVM e tratamento SVD para as características de coloração e volume (próprio autor).

SVM – Tratamento por SVD						
Nº de amostras	Acurácia ao próprio treinamento			Acurácia ao teste		
	Coloração (%)	Volume (%)	Tempo	Coloração (%)	Volume (%)	Tempo
40	100	100	22s	81,25	90,62	32s
80	100	100	44s	81,25	95,31	32s
120	100	100	1min10s	81,25	95,31	32s

Em seguida, para o tratamento PCA, foram obtidos os resultados apresentados na Tabela 6 e que também serão abordados posteriormente.

Tabela 6 - Relação de acurácia (%) no treinamento por SVM e tratamento PCA para as características de coloração e volume (próprio autor).

SVM – Tratamento por PCA						
Nº de amostras	Acurácia ao próprio treinamento			Acurácia ao teste		
	Coloração (%)	Volume (%)	Tempo	Coloração (%)	Volume (%)	Tempo
40	100	100	21s	81,25	90,62	33s
80	100	100	44s	81,25	95,31	33s
120	100	100	59s	81,25	95,31	32s

Os resultados para o tratamento por segmentação da imagem colorida e treinamento por SVM podem ser observados na Tabela 7. Foram obtidos valores interessantes para a saída de análise da coloração e do volume onde, para o treinamento com 120 amostras, foi obtida apenas uma previsão errada em relação às 64 presentes no teste.

Tabela 7 - Relação de acurácia (%) no treinamento por SVM e tratamento de segmentação para as características de coloração e volume (próprio autor).

SVM – Tratamento por Segmentação						
Nº de amostras	Acurácia ao próprio treinamento			Acurácia ao teste		
	Coloração (%)	Volume (%)	Tempo	Coloração (%)	Volume (%)	Tempo
40	100	100	1min10	60,94	93,75	1min53
80	86,25	100	2min20	75	98,44	1min53
120	82,5	100	3min30s	78,12	98,44	1min53s

Se tratando dos resultados apresentados nas tabelas 5 e 6, é possível notar a persistência do valor 81,25%. Este resultado não é considerado aceitável, uma vez que representa exatamente o número de amostras não transparentes presentes no banco de dados. Se tratando da classificação em relação à coloração, para amostras transparentes, é dado o valor 1 e para as turvas ou coloridas, é dado o valor 0. O que acontece neste caso é que o algoritmo retorna como resposta para todos os casos o valor 0. Ou seja, ele acredita que todas as amostras são inválidas. Devido ao número de amostras com valor 0 ser consideravelmente maior que o número de amostras com valor 1, o algoritmo não teve informação suficiente para aprender a diferença entre as possíveis respostas, acreditando que, em todos os casos, não há refil transparente. De forma geral, resultados de coloração para as imagens de teste com valores iguais ou menores que 81,25% podem ser descartados. Apenas resultados acima deste valor podem ser considerados como tendo alcançado um mínimo de aprendizado para classificação do problema proposto. Assim sendo, o algoritmo RNA com tratamento por SVD e PCA obteve melhor resultado na classificação da coloração, com valores de 82,81% e 92,19%, respectivamente.

Tratando do problema de *overfitting*, com base nos testes realizados, observou-se que algumas abordagens podem ser tomadas, como o aumento do banco de dados priorizando a utilização de imagens diferentes e a redução do tamanho da rede neural artificial.

Para o caso da classificação de volume e coloração, os testes realizados pelo método SVM apresentaram maior valor de acurácia, chegando a 98,44%.

Foi desenvolvida, também, uma única estrutura de classificação empregando o método RNA em que houvesse como entrada todos os parâmetros para coloração e todos para volume, além de uma aplicando o método SVM. Neste caso, ao invés de possuir dois treinamentos diferentes para a coloração e para o volume, esta estrutura tratava de ambas as características em um único treinamento. A melhor acurácia encontrada para estes casos, aplicando os três tipos de tratamento de dados, foi de 96,875%.

Também foi desenvolvida uma estrutura aplicando o método RNA, inicialmente, para classificação da coloração, em série à uma classificação de volume pelo método SVM, onde as entradas desta segunda seriam os parâmetros para definição do volume e a saída da classificação da coloração, de forma a obter como resposta 1 todo refil com volume e coloração ideais e 0 qualquer saída diferente desta. Para este caso, foram utilizados os parâmetros de melhor resultado para a coloração, que foi o caso do algoritmo RNA com pré-processamento por PCA e banco de dados de treinamento de 120 amostras, e um dos parâmetros de melhor resultado para o volume, que foi o SVM, também com pré-processamento por PCA e banco de dados de treinamento de 120 amostras. Ao aplicar esta estrutura em série, foi obtida acurácia de 100% ao banco de dados de testes de 64 amostras, classificando todos os refis corretamente.

## 5 CONCLUSÕES

Um ponto importante que não foi possível tratar durante o desenvolvimento deste trabalho foi a implementação de identificação de deformidades em tempo real, através da utilização de câmeras. É uma opção que fica em aberto para uma possível pesquisa futura.

Conclui-se que foi obtido um resultado satisfatório. Para um primeiro projeto acerca do assunto, foi de grande valia, uma vez que foi um estudo multidisciplinar, não se limitando apenas ao campo das redes neurais artificiais e aprendizagem de máquinas, mas também abrangendo métodos de pré-processamento de dados, redução de matrizes de características e programação em linguagem Python voltada à GUI.

## REFERÊNCIAS

Abreu, R. A. *Perdas no processo produtivo*. RAA Consultoria: 2002.

Araújo Júnior, Antônio P. de; CHAGAS, Christiano V. de; FERNANDES, Raphaela G. *Uma rápida análise sobre automação industrial*. Redes para Automação Industrial. DCA-CT-UFRN. 2003.

Ferg, Stephen. *Thinking in Tkinter*. 2009. Disponível em: <http://thinkingtkinter.sourceforge.net/>

HUMULIN®: *Insulina humana (derivada de ADN\* recombinante)*. Indústria Lilly France S.A.S. Distribuído por Eli Lilly do Brasil Ltda. CRF-SP 11422. Bula de remédio on-line. Disponível em: <http://www.saudedireta.com.br/catinc/drugs/bulas/humulin.pdf>

Louro, Antônio Henrique Figueira; MENDONÇA, Michelle Magalhães; GONZAGA, Adilson. *Classificação de tomates utilizando redes neurais artificiais*. Anais. São Carlos: EESC-USP, 2006.

NOVORAPID® FLEXPEN®: *insulina asparte*. Fabricado por: Novo Nordisk A/S DK-2880, Bagsvaerd, Dinamarca. Importado por: Novo Nordisk Farmacêutica do Brasil Ltda. Farmacêutico responsável: Luciane M. H. Fernandes CRF/PR n°6002. Esta bula foi aprovada pela Anvisa em 17/03/2017. Disponível em: <http://www.novonordisk.com.br/content/dam/brazil/affiliate/www-novonordiskbr/Bulas/NovoRapid%20FlexPen%20Paciente.pdf>

Robles Júnior, A. *Custos de qualidade; uma estratégia para a competição global*. São Paulo: Atlas. 1994.

Stivanello, Maurício. E. *Inspeção Industrial através de Visão Computacional*. Trabalho de Conclusão de Curso do curso de Bacharel em Ciências da Computação, Universidade Regional de Blumenau. 2004.