



## SISTEMA DE INSPEÇÃO DE DEFEITOS EM PERFIS DE ALUMÍNIO UTILIZANDO A VISÃO COMPUTACIONAL

**Alexandre da Silva Rufino** – asr.projetoscad@gmail.com

**Newton Spinoza Sobrinho** – newtonespinoza@hotmail.com

**Charles Luiz Silva de Melo** – charles.melo@yahoo.com.br

**Marivan Silva Gomes** – marivan.gomes@gmail.com

Universidade do Estado do Amazonas, Escola Superior de Tecnologia

Av. Darcy Vargas, 1200

69055-035 – Manaus – Amazonas

**Resumo:** Atualmente a inspeção visual é largamente difundida nos mais variados domínios, permitindo a identificação de dados de maneira mais eficaz, pois consegue eliminar ou reduzir os erros ocasionados pelo cansaço ou fadiga, por exemplo. A área é dividida em: visão computacional e processamento de imagens. Este trabalho se propôs a estudar um método de inspeção de furação em perfis de alumínio utilizando técnicas de processamento de imagens digitais em visão computacional, realizando a verificação de presença, posicionamento e diâmetro dos furos. O algoritmo criado permite inspecionar a qualidade de furos de perfis de alumínio por intermédio de funções do módulo Vision Assistant do software LabVIEW®.

**Palavras-chave:** Processamento digital de imagens, visão computacional, LabView, inspeção de qualidade.

### 1. INTRODUÇÃO

Tentar emular a inteligência humana ou criar mecanismos que se aproximem disso não é algo recente. Além de técnicas que substituíssem ou reduzissem o esforço físico, o homem também sempre buscou meios para substituir ou auxiliar o raciocínio lógico e a inteligência humana. Desde o ábaco e o relógio solar até os computadores atuais, a criação desses mecanismos sempre foi um campo tido como prioritário no estudo e pesquisa humanos.

Utilizando técnicas de modelagem de sistemas dinâmicos em um cérebro humano, obtêm-se os estímulos sensoriais como dados de entrada, a interpretação e processamento desses dados como plantas e controladores internos e como saída, pode-se ter informações processadas e/ou tomadas de decisões.

Partindo dessa analogia, podemos definir a visão computacional como um conjunto de métodos e técnicas através dos quais sistemas computacionais são capazes de interpretar e extrair dados relevantes de imagens (GONZALEZ, 2008). Modelando novamente, temos a imagem como entrada e a informação como saída.

A visão computacional, objeto de estudo desse trabalho, vem sendo foco de um grande número de pesquisas ao longo das últimas décadas. Entretanto, ainda hoje, não existe uma formulação padrão para os problemas que a envolvem. O que existem são métodos para resolver tarefas bem definidas, pré-programadas e com baixa versatilidade. Uma das grandes dificuldades na detecção de padrões é a grande variação na aparência de objetos tidos como “iguais” sob diferentes condições, como ângulo de captura da imagem, foco, iluminação,



definição de imagem e filtros aplicados (Zhang, 2006). Um sistema de Visão Computacional só pode ser considerado eficiente se conseguir eliminar ou reduzir essas variações, criando um padrão uniforme para análise.

A indústria, por empregar atividades padronizadas e repetitivas em sua produção em massa, não tem essas limitações como um empecilho e atualmente faz grande uso da visão computacional no controle do processo produtivo, sobretudo na área de inspeção de qualidade, que requer essa padronização supracitada.

A qualidade de produção de blocos de alumínio da empresa fictícia Espina S/A é de vital importância, já que o produto é o carro-chefe dos produtos feitos por ela. Uma variação ou defeito percebidos por um cliente mais atento, apesar de por vezes não ser crítico para a funcionalidade do aparelho, pode acarretar em uma devolução ou troca e - o que é mais crítico para a empresa, posto que é incomensurável - minar a confiança no produto e na empresa.

A inspeção humana está sujeita a uma série de interferências. Fadiga, iluminação da área de inspeção, acuracidade da visão do inspetor, diferenças de interpretação de defeitos e até mesmo o humor do inspetor podem entrar na balança na hora de inspecionar uma peça.

O trabalho da visão computacional é eliminar ou ao menos mitigar esses ruídos externos que podem influenciar a inspeção de qualidade.

## 2. METODOLOGIA

Para um melhor entendimento do projeto abordado nesse trabalho, foi necessário um estudo sobre os conceitos básicos de processamento digital de imagens. A introdução teórica do capítulo 2 é o fruto desse estudo.

A etapa seguinte do trabalho foi o estudo sobre as técnicas, métodos e ferramentas utilizados no processamento digital de imagens. Por orientação do professor Marivan Gomes, optou-se pelo módulo de visão computacional Vision Assistant do software LabVIEW®, da empresa National Instruments.

O **LabVIEW®**, abreviação de *Laboratory Virtual Instrument Engineering Workbench*, é uma linguagem de programação gráfica que surgiu em 1986, desenvolvido pela empresa *National Instruments*. Originalmente para sistemas Macintosh, hoje conta com versões para Sistemas Operacionais Windows, Linux e Solaris.

Os principais campos de aplicação do LabVIEW são a realização de medições e a automação de processos. A programação é feita de acordo com o modelo de fluxo de dados e diagrama de blocos, o que oferece a esta linguagem vantagens para a aquisição e manipulação de dados.

Os programas em LabVIEW são chamados de instrumentos virtuais ou, simplesmente, IVs (VI em inglês). São compostos pelo painel frontal, que contém a interface, e pelo diagrama de blocos, que contém o código gráfico do programa. O programa não é processado por um interpretador, mas sim compilado. Assim o seu desempenho é comparável à exibida pelas linguagens de programação de alto nível. A linguagem gráfica do LabVIEW é chamada "G".

Os blocos de funções são designados por instrumentos virtuais. Em princípio, cada programa (Sub-IV) pode ser usado como sub-programa por qualquer outro programa ou pode, simplesmente, ser executado isoladamente. Devido à utilização do modelo do fluxo de dados, as chamadas recursivas não são possíveis, podendo-se, no entanto, conseguir esse efeito pela aplicação de outras técnicas.



O programador liga IVs com linhas (*wires*) de ligação e define, deste modo, o fluxo de dados. Cada IV pode possuir entradas e/ou saídas. A execução de um IV começa quando todas as entradas estão disponíveis. Os resultados do processamento são então colocados nas saídas assim que a execução do sub-programa tenha terminado. Desta forma, a ordem pela qual as tarefas são executadas é definida em função dos dados.

Uma importante consequência destas regras é a facilidade com que podem ser criados processos paralelos no LabVIEW. Os sub-IVs sem interdependência dos respectivos dados são processados em paralelo. Em projetos de grande dimensão é muito importante planejar a sua estrutura desde o início (como acontecem nas outras linguagens de programação).

Os sub-IVs que não possuem entradas são executados no início do programa. Se o sub-IV não possuir saídas, os dados resultantes são ignorados ou, então, usados externamente: são escritos para o disco rígido ou para a rede, ou enviados para impressão. Da mesma forma, um sub-IV sem entradas pode receber dados provenientes de aparelhos periféricos ou pode gerar os seus próprios dados (um exemplo é um gerador de números aleatórios). No caso desse trabalho, foi utilizada uma webcam como fonte de dados de alimentação de entrada e *leds* indicadores como saída.

O painel frontal do LabVIEW é a sua interface gráfica para o usuário final. O programador não necessita escrever qualquer linha de código. A apresentação gráfica dos processos aumenta a facilidade de leitura e de utilização. Uma grande vantagem em relação às linguagens baseadas em texto é a facilidade com que se criam componentes que se executam paralelamente.

Por essa facilidade de utilização, optou-se pela utilização do LabVIEW na realização desse trabalho, preterindo outras ferramentas mais tradicionais do ramo da engenharia.

De acordo com o site da National Instruments, que fala sobre o módulo *Vision Development*:

*O módulo Vision Assistant Development é uma ferramenta para prototipagem e teste de aplicações de processamento de imagens. Ao prototipar uma aplicação de processamento de imagens utilizando diagramas de blocos, é criado um algoritmo que registra todos os passos da aplicação. Esse algoritmo é gravado em um arquivo de texto ASCII que lista as funções de processamento e parâmetros relevantes para a prototipagem. Ao término da criação do algoritmo, o NI Vision gera, através de um assistente, uma VI de interface com o usuário no LabVIEW®.*

Posteriormente, definiram-se as etapas do ciclo de inspeção de qualidade para que o objetivo final da aplicação fosse atingido. Essas etapas incluem o posicionamento da peça no *jig*, a leitura da peça pelo sistema de visão computacional e a aprovação ou não da peça. Para testar a aplicação, foram desenhadas (utilizando o software Autodesk Inventor 13) e confeccionadas 20 amostras de perfis de alumínio, sendo 4 representando peças-padrão dentro da conformidade e 16 com diferentes tipos de defeitos para inspeção de qualidade:

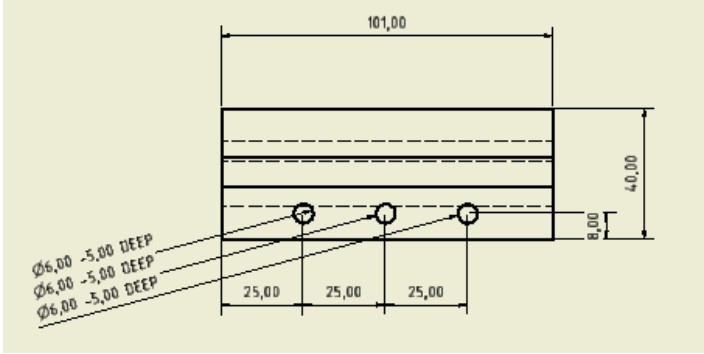


Figura 1 – Desenho de amostra de peça

Peças confeccionadas:



Figura 2 – Peças confeccionadas

Foi também confeccionado um *jig* para fixação das peças e da câmera de inspeção:

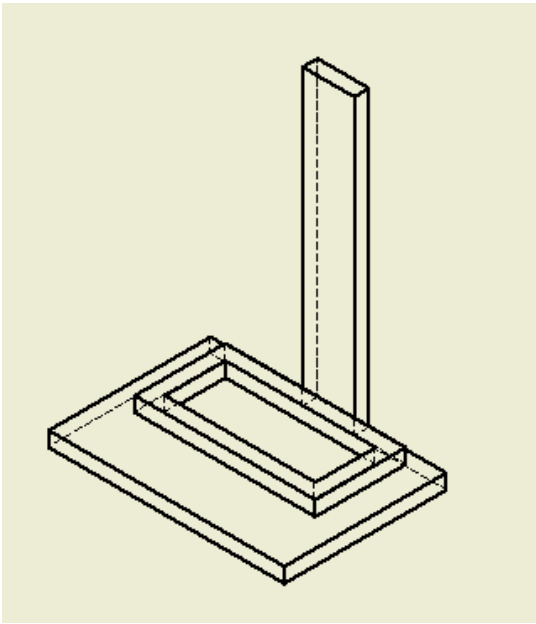


Figura 3 – JIG de fixação

*Jig* confeccionado:



**Figura 4** – *JIG* confeccionado

Por fim, foi criado o algoritmo LabVIEW® que realiza as tarefas de inspeção da peça por visão computacional. Essas tarefas incluem a aquisição da imagem, segmentação de cores, conversão de medidas de pixels para o sistema métrico, definição de área de visão computacional, obtenção das medidas desejadas para inspeção e criação de uma saída indicadora de sucesso ou falha.

Nesse tópico foram analisados os resultados obtidos após o experimento, a sua utilidade na indústria e a sua relevância enquanto amostra de um sistema de automação.

### **3. PROCESSAMENTO DIGITAL DE IMAGENS**

O estudo de visão computacional ainda pode ser considerado jovem e imaturo, uma vez que só a partir dos anos 70, quando os computadores passaram a processar grandes conjuntos de dados – como imagens – é que foram possíveis estudos mais aprofundados e específicos sobre o tema (Magalhães, 2011).

Ainda hoje não existe uma formulação padrão de como resolver os problemas envolvendo a visão computacional. Utilizam-se diversos métodos e técnicas para realizar um conjunto muito específico de tarefas bem definidas. A soma de toda essa gama de processamentos e análises gera o resultado final ou uma tarefa em particular (Zhang, 2006).

Mesmo com alguns empecilhos operacionais, podemos perceber que hoje a visão computacional já se encontra vastamente inserida nos mais variados campos de estudo moderno.

A inteligência artificial a utiliza para tomadas de decisões autônomas em sistemas que requerem a movimentação de um robô em um ambiente espacial real (Magalhães, 2011). Conforme disposto anteriormente, a modelagem desses sistemas se baseia na captura de informações sensoriais (visão) de entrada, alimentando o sistema com dados qualitativos



sobre o sistema. Esses dados são posteriormente processados e geram saídas convertidas em tomadas de decisões.

Sob esse foco de estudo, utiliza-se a visão computacional, por exemplo, na agricultura de precisão (FALECO, 2012). Torna-se possível a navegação baseada na visão tridimensional, navegação noturna, manejo correto de linhas de extração e controle de plantas daninhas mais próximas à planta extraída.

Outro campo que vem sendo largamente explorado e onde a visão computacional é de vital importância é na chamada Realidade Aumentada, que é a integração de informações virtuais a visualizações do mundo real (como, por exemplo, através de uma câmera) (Thomas, 2000). Atualmente, a maior parte das pesquisas em RA está ligada ao uso de vídeos transmitidos ao vivo, que são digitalmente processados e “ampliados” pela adição de gráficos criados pelo computador. Pesquisas avançadas incluem uso de rastreamento de dados em movimento, reconhecimento de marcadores confiáveis utilizando mecanismos de visão, e a construção de ambientes controlados contendo qualquer número de sensores e atuadores.

Concluindo o estado da arte do campo da Visão Computacional, temos uma das tecnologias de maior sucesso, eficácia e versatilidade atualmente, que é o sistema Kinect™ da Microsoft. O Microsoft Kinect™ (MICROSOFT SDK, 2011) é um sensor de movimento associado com um sistema de processamento de imagens, lançado em novembro de 2010. Ele é capaz de reconhecer e mapear pontos do corpo humano que são unidos, formando juntas que geram uma espécie de esqueleto. Isso pode ser visto na figura:

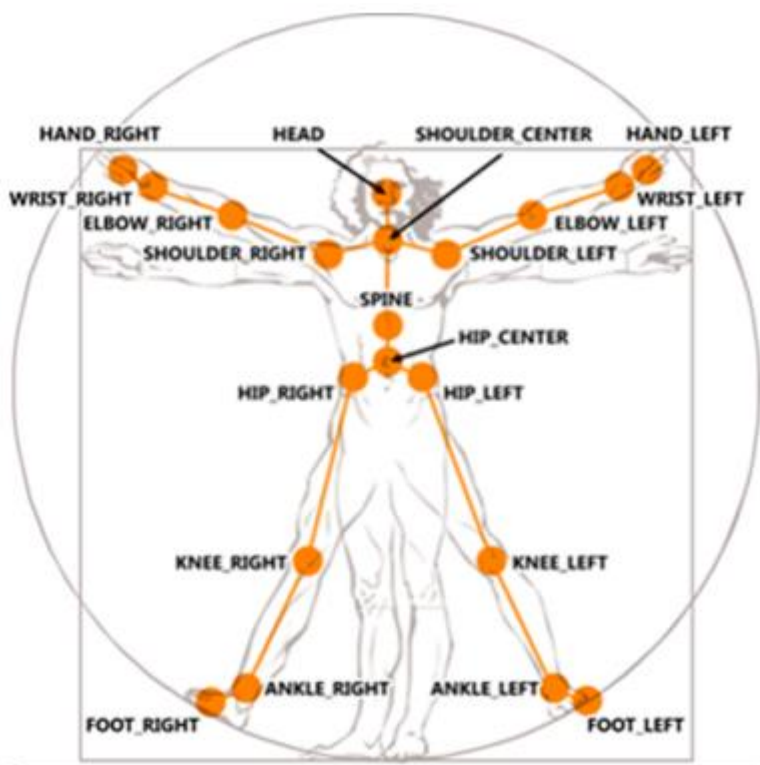


Figura 6 – Esqueleto do Kinect

Apesar de a função principal do dispositivo seja de servir de interface para jogos eletrônicos, ele vem sendo utilizado nos mais variados campos, tais como a biotecnologia, sistemas de segurança, navegação, etc.

Outro exemplo de aplicação da Visão Computacional no controle de qualidade é o da indústria madeireira (OLIVEIRA, 2008).

Um problema comum na seleção de madeiras de pinho é a existência de nós. A existência ou não de nós visíveis na madeira faz com que tábuas sejam classificadas em diferentes categorias de qualidade. Abaixo podemos ver como uma solução bastante simples utilizando uma limiarização baseada em histograma detecta nós em madeiras de pinho.



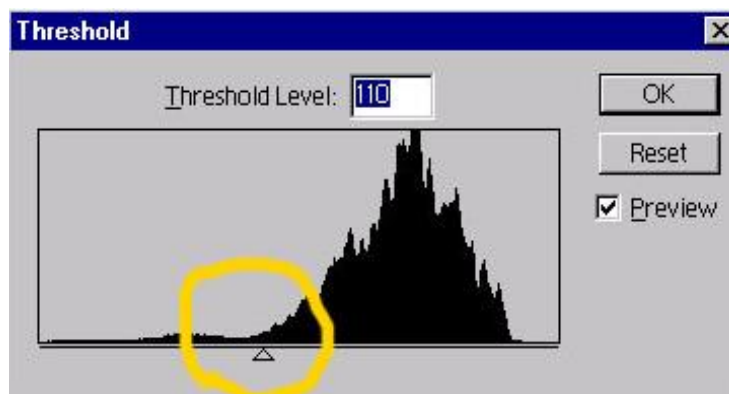
**Figura 7** – Nós de pinho

A imagem acima mostra uma tábua de pinho com nós.



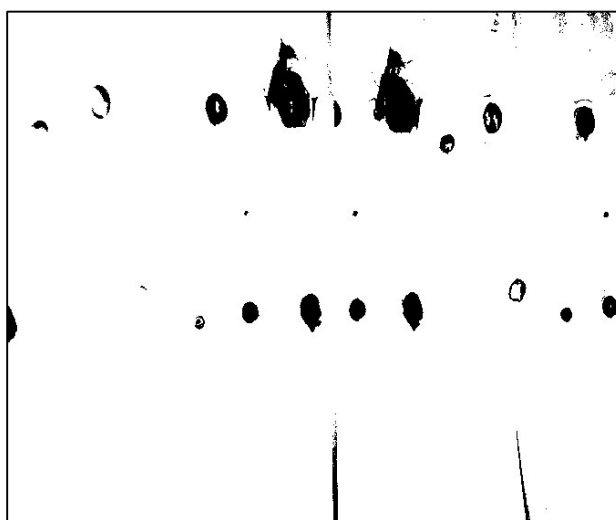
**Figura 8** – Nós de pinho em escala de cinza

A mesma imagem anterior, mas em escala de cinza.



**Figura 9** - Histograma

Acima vemos o histograma da imagem. Supondo que nós são ocorrências raras e mais escuras, realiza-se uma varredura no histograma da imagem e a partir dos pixels de menor valor e nota-se onde começa a maioria dos valores da imagem. Supondo que os valores escuros encontrados em menor quantidade representam falhas, define-se um limiar em pixels de valor = 110 (para uma faixa de 0 a 255).



**Figura 10** – Nós de pinho com borda

Acima vemos a imagem limiarizada em 110. Os pontos escuros ressaltados são os nós. Utiliza-se um algoritmo para encontrar agrupamentos redondos e escuros na imagem, podendo assim determinar quais tábuas estão dentro da conformidade de qualidade.

### 3.1 Imagem Digital

Uma imagem digital pode ser vista como uma matriz cujas linhas e colunas identificam um ponto na imagem, cujo valor corresponde ao nível de cinza da imagem naquele ponto (MARQUES FILHO, 2011).

8	6	3
3	8	4
0	6	2

**Figura 11** – Matriz com tons de cinza

Ela é adquirida através de um Dispositivo de Carga Acoplada ou CCD (*charge-coupled device*), que é um sensor para captação de imagens formado por um circuito integrado que contém uma matriz de capacitores acoplados (FERNANDES, 2012).





Essa matriz é constituída por células fotoelétricas que retém a informação de quantidade de luz que lhes atinge, registrando o número de fótons integrado no tempo.

A organização em matriz retangular permite identificar a localização espacial das várias células e posteriormente quantificar a intensidade de luz armazenada individualmente

### 3.2 Formação da Imagem Digital

Existem 3 fases essenciais no processo de transformação de um sinal (aqui, no caso, uma imagem) analógico em digital: amostragem, quantização e codificação.

O processo de **amostragem** consiste em obter, dentro do domínio dos valores dos fótons, valores para representar a função naquele trecho. Em suma, apenas um valor é considerado em cada célula.

Após a amostragem temos a **quantização**. O valor amostrado pode ser adquirido como um ponto flutuante ou inteiro longo sem um valor limite, podendo requerer mais memória do que podemos dispor. Para reduzir o tamanho da representação do número e o uso de memória, dividimos os valores adquiridos na amostragem em classes e escolhemos um representante para cada classe.

A **codificação** estabelece quais cadeias de bits representam uma determinada cor ou nível de cinza, mais precisamente estabelece os bits que representam os níveis dos tons de cinza ou dos canais de cores.

Esse processo transforma cada célula em unidades de imagens ou *picture elements* (pixels).

### 3.3 Pixel

O pixel é o menor componente de uma imagem digital (FERNANDES, 2012). É um elemento da matriz que contém um valor. Esse valor define a intensidade da luz ou da cor naquele ponto. É chamado de profundidade de bit e depende do número de bits disponíveis para armazená-lo:

- 1 bit (0 ou 1) – 2 cores. Preto e branco;
- 2 bits = 4 cores;
- 4 bits:  $2^4 = 16$  cores;
- 8 bits:  $2^8 = 256$  cores;
- 24 bits:  $2^{24} = 16.8$  milhões de cores.

### 3.4 Profundidade de bit

O número de bits utilizados para descrever um pixel é conhecido como sua profundidade em bit (*bit depth*). A *Profundidade de bits* especifica quantas informações sobre cores estão disponíveis para cada pixel da imagem. Quanto mais bits de informação por pixel, mais cores disponíveis haverá e mais precisa será a representação das cores. Por exemplo, uma imagem ou uma profundidade de bits igual a 1 tem pixels com dois valores possíveis: preto e branco. Uma imagem com profundidade de bits igual a 8 tem 28 ou 256 valores possíveis. As imagens no modo Tons de cinza com profundidade igual a 8 têm 256 valores de cinza possíveis (BONVETI, 2006).

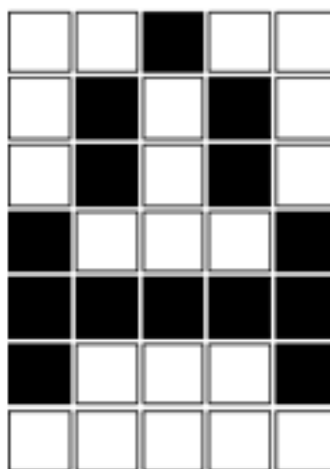
As imagens digitais coloridas são representadas como um vetor de três componentes (Gattass, 2002), onde cada componente varia de 0 a 255 de acordo com a cor. Os modelos são:

- **Modelo RGB** (*Red, Blue e Green*): formado pelas cores vermelho, verde e azul;
- **Modelo CMY** (*Cyan, Magenta e Yellow*): modelo RGB com cores secundárias;
- **Modelo HSI** (*Hue, Saturation e Intensity*): formado por uma cor dominante (matiz), saturação (diluição da matiz na luz branca) e intensidade (quantidade de luz refletida).

### 3.5 Tipos de Imagem

Existem basicamente dois tipos de imagens digitais: Mapa de bits (bitmap) e imagens vetoriais. Nos mapas de bits ou imagens raster, cada ponto da matriz armazena uma cor e corresponde a um ponto da imagem. O número de bits para armazenar cada ponto pode variar determinando a quantidade de cores da imagem.

É o meio eletrônico mais comum para armazenamento de imagem de tons contínuos, como fotografias ou pinturas, pois representam gradações sutis de cor e sombra. As imagens bitmap são dependentes da resolução, isto é, contém um número fixo de pixels e podem sofrer deformações ao serem redimensionadas.



**Figura 12** - Bitmap

Nas imagens vetoriais, são armazenados posições e tamanhos dos traços. É um tipo de imagem gerada a partir de descrições geométricas de formas. Uma imagem vetorial normalmente é composta por curvas, elipses, polígonos e outros elementos paramétricos, utilizando vetores matemáticos para sua descrição.

As Curvas de Bézier são usadas para a manipulação dos pontos de um desenho. Cada linha descrita em um desenho vetorial possui nós, e cada nó possui alças para manipular o segmento de reta ligado a ele.

Por serem baseados em vetores, esses gráficos geralmente são mais leves (ocupam menos espaço em mídias de armazenamento) e não perdem qualidade ao serem ampliados, já que as funções matemáticas adequam-se à escala, o que não ocorre com gráficos *bitmap*, que utilizam métodos de interpolação na tentativa de preservar a qualidade. Outra vantagem do desenho vetorial é a possibilidade de isolar objetos e zonas, tratando-as independentemente.

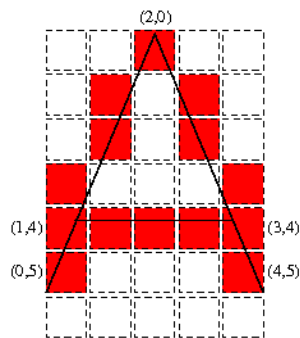


Figura 13 - Vetorial

### 3.6 Histograma

Em processamento de imagens, trabalha-se sempre com os tons de cinza atribuídos aos pixels de uma imagem (SILVA, 2001). O histograma de uma imagem fornece a informação sobre quantos pixels na imagem possuem cada valor possível de cinza (que, no caso das imagens de 8 bits, variam de 0 a 255) ou, de forma equivalente, qual a proporção da imagem que corresponde a cada valor de cinza. Outro ponto importante com relação a histogramas é que eles representam dados digitais, também chamados de discretos. Assim sendo, a distribuição de intensidades é representada por colunas discretas, que não podem ser divididas ou "quebradas", correspondentes a números inteiros (em contraposição a números fracionários). Esse conceito assume importância ao se tratar de realce de contraste em imagens. Ao se observar o histograma de uma imagem, tem-se uma noção instantânea sobre as características da mesma. A forma do histograma fornece informações de grande importância no caso das imagens de sensoriamento remoto, tais como intensidade média e espalhamento dos valores de cinza; este último, por sua vez, dá a medida do contraste de uma imagem:

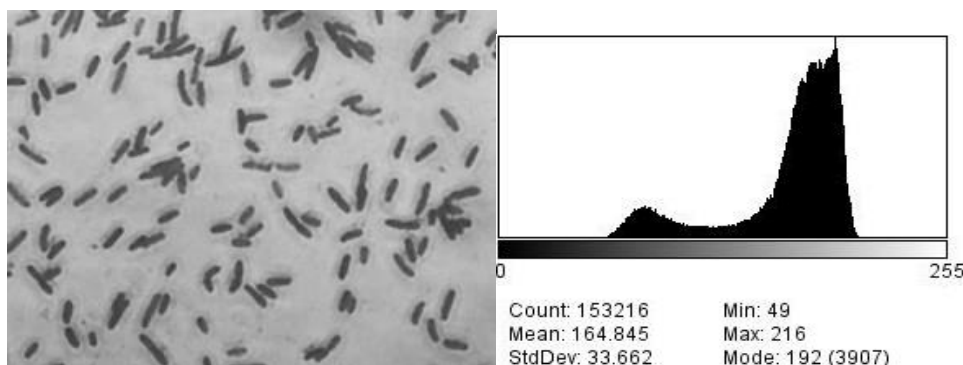


Figura 14 – Imagem com histograma

Segmentando o RGB da imagem acima, obtemos o tom de cinza e analisamos o seu histograma.

## 4. CONCLUSÃO

Os sistemas de visão computacional podem ser aplicados em diversas áreas, principalmente onde a permanência do ser humano pode representar algum risco a saúde ou segurança. Em sistemas industriais o interesse é maior por conta da redução de custos, pois o



mercado competitivo busca uma qualidade melhor com um custo menor, visando o aumento da produtividade e a satisfação do cliente.

Com os resultados obtidos, concluiu-se que um modelo computacional pode auxiliar e até mesmo substituir um sistema de inspeção de qualidade. Também ao término do trabalho foi adquirido um conhecimento maior a respeito da utilização do módulo Vision Assistant, presente na ferramenta *LabVIEW*®.

Com o código validado pelas simulações, a implantação do sistema se torna possível, faltando apenas o estudo completo da parte mecânica do projeto na linha de produção ou na linha de montagem da peça, com equipamentos de maior qualidade e iluminação adequada.

Portanto, os resultados deste trabalho foram satisfatórios, podendo ser implementado em empresas que possuem sistemas de inspeção. Lembrando que o algoritmo de análise é elaborado para o uso específico e de acordo com os dados de uma determinada peça, e as técnicas de reconhecimento variam de caso a caso.

## 5. REFERÊNCIAS

ALBUQUERQUE, Pedro Urbano Braga; ALEXANDRIA, Auzuir Ripardo. **Redes Industriais**. São Paulo: Ensino, 2009.

FRANCHI, Claiton Moro; CAMARGO, Valter Luís Arlindo de. **Controladores Lógicos Programáveis - Sistemas Discretos**. São Paulo: Editora Érica. 2008.

MAMEDE FILHO, João. **Instalações elétricas industriais**. 6 ed. Rio de Janeiro: LTC, 2001.

MORAES, Cícero Couto de; CASTRUCCI, Plínio de Lauro. **Engenharia de automação industrial**. 2 ed. São Paulo: LTC, 2007.

SEIPPEL, Robert G. **Transducers, sensors & detectors**. [S.I]: [S.ed.], 1983.

SILVA, Daniel Soares. Apostila Criativa Automação Industrial S.A. **Curso Básico de PLC**. São Paulo: ITA, 2006.

SILVEIRA, Paulo Rogério da; SANTOS, Winderson E. dos. **Automação e controle discreto**. 4. ed. São Paulo, SP: Érica, 2002.

## DEFECT INSPECTION SYSTEM IN ALUMINUM PROFILES USING A COMPUTER VISION

**Abstract:** *Currently, visual inspection is widespread in various work fields, allowing the identification of data more effectively, since it can eliminate or reduce errors caused by fatigue or tiredness, for example. The area is divided into: computer vision and image processing. This work proposes a study about a method of inspection of holes on aluminum profiles using techniques of digital image processing in computer vision, checking the presence, position and diameter of the holes. The algorithm allows to inspect the quality of the aluminum profiles holes using functions of LabVIEW's module, Vision Assistant.*

**Key-words:** *Computer Vision, Image Processing, LabView, quality control.*