

CONTROLE DE ESTOQUE BASEADO EM RECONHECIMENTO DE PADRÃO DE IMAGEM

Eric Vianna Kloeble - eric.kloeble@gmail.com

Lucas Larrubia Borgonovi – lucasborgo9@icloud.com

Bruno Luis Soares de Lima (Orientador) - bruno.lima@mackenzie.br

RESUMO

O presente trabalho refere-se ao uso de tecnologias de reconhecimento de padrões de imagem para a identificação de rótulos e etiquetas de produtos em um armazém industrial e contabilização. Imagens captadas através de uma câmera acoplada a um veículo remotamente pilotado serão analisadas através de uma central de processamento com um algoritmo em linguagem de programação Python que enviará as informações a um sistema de monitoramento remoto.

Palavras-chave: visão computacional. Inventário. Programação. Drone. OCR. OpenCV.

INVENTORY CONTROL BASED ON IMAGE PATTERN RECOGNITION

ABSTRACT

The present work refers to the use of image pattern recognition technologies for the identification of product labels in an industrial warehouse and their quantity. Images captured through a camera coupled to a remotely piloted vehicle are analyzed through a processing center with a Python programming language algorithm which sends the information to a remote monitoring system.

Keywords: Computer vision. Inventory. Programming. Drone. OCR. OpenCV.

1 INTRODUÇÃO

O mercado varejo movimentou na área de logística de acordo com Market and Market (2020), em 2020 cerca de 2,734 bilhões de dólares, neste mesmo ano, foram investidos 48,4 bilhões voltados em processos automatizados nesse setor (MARKETS AND MARKETS, 2021). Mostrando a grande preocupação das empresas com a aplicação na visão de negócios da transformação digital.

A transformação digital de acordo com Pihir, et al (2018) se baseia na mudança de infraestrutura e processos em organizações que se apoiam em tecnologias de informação e comunicação para a criação de novos modelos de negócio. Permitindo alcançar maior produtividade, flexibilidade, qualidade e gerenciamento de negócio (SACOMANO et al, 2018).

Dentro de um sistema de cadeia logística, uma parte importante é o gerenciamento de estoque. Ele envolve não apenas o movimento de entrada e saída de mercadorias, mas várias etapas que devem ser realizadas como a análise de dados de compra, de novos pedidos, expedição, armazenamento, entre outras.

Dentre essas etapas, a gestão do armazenamento é crucial para garantir que a empresa atenda às necessidades de seus clientes ao mesmo tempo que evita falhas nas outras etapas do processo.

Esta gestão é uma tarefa árdua feita através da contagem dos itens em estoque, porém muito necessária pois possibilita tanto o total conhecimento da quantidade exata de produtos disponíveis como um melhor planejamento para aqueles que precisam ser encomendados.

Essa contagem pode demandar a pausa de outras atividades exigindo rapidez para reduzir o custo dessa interrupção. Aloca-se na maioria das vezes colaboradores para realizar a apuração, porém devido ao ritmo acelerado pode gerar uma taxa de erro considerável.

O avanço da tecnologia possibilitou diversas alternativas para facilitar o processo de gestão de estoques, aumentando a taxa de assertividade em

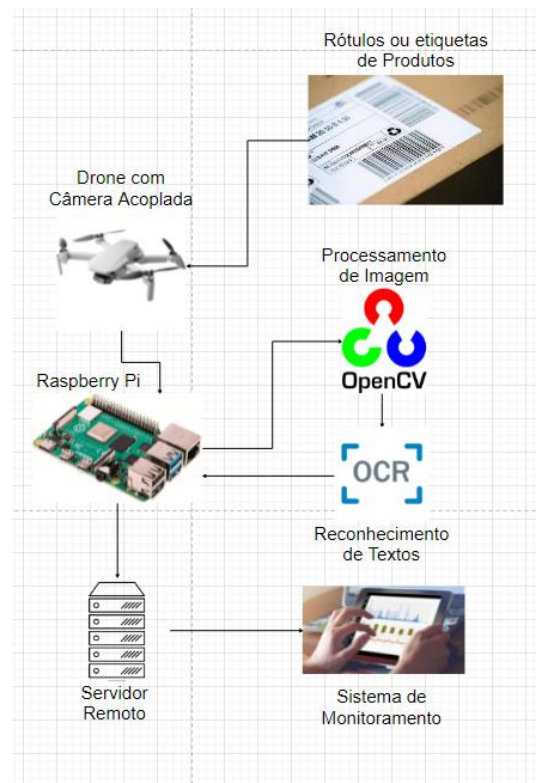
aproximadamente 18% de acordo com Aires (2019 apud Min et al, 2010) e diminuindo o tempo necessário para realização de um inventário. Algumas tecnologias como a inteligência artificial (SILVA; BROTTI; CASTRO, 2014), utilização sensores como RFID (ARYAL, 2012), Drones (WAWRLA; MAGHAZEI; NETLAND, 2019), leitores de códigos de barras (HONG-YING, 2009), já estão sendo aplicadas em alguns cenários de gestão de estoques.

Neste cenário de crescente automatização de estoques e da cadeia logística, o presente projeto propõe uma aplicação de reconhecimento de padrão de imagem para desenvolvimento de um sistema embarcado num drone para identificação de objetos presentes num estoque.

O projeto teve como objetivo facilitar o processo de contagem de produtos no estoque por meio de um sistema a ser desenvolvido que contará com uma câmera acoplada a uma aeronave remotamente pilotada (RPA) que terá embarcado um microprocessador executando um algoritmo de processamento de imagens.

A figura 1 apresenta a arquitetura do sistema proposto neste trabalho, onde, o drone se locomove com o Raspberry e sua câmera por todo o galpão através de comandos remotos de um operador capacitado, que será responsável por fotografar cada produto. Cada foto passa por dois algoritmos, um para o seu processamento e outro para o reconhecimento dos textos de interesse. Por fim, as informações serão enviadas a um banco de dados Cosmos DB pela internet.

Figura 1 – Arquitetura do sistema



Fonte: Os Autores (2021)

1.1 Objetivos

1.1.1 Objetivo Geral

Desenvolver um sistema baseado em reconhecimento de padrão de imagens que permita identificar etiquetas e rótulos de caixas de produtos em armazéns e realizar a correta contabilização de sua quantidade.

1.1.2 Objetivos específicos

- Especificar um processador capaz de executar o algoritmo a ser desenvolvido para coleta e reconhecimento de imagens;
- Desenvolver um algoritmo para coleta e reconhecimento de imagens de etiquetas e rótulos através de uma câmera acoplada ao *drone*.
- Desenvolver a integração para envio das informações de contagem e identificação dos objetos via rede de comunicação para um servidor.

1.2 Justificativa

Como destacado por Oliveira e seus colaboradores (2011), é importante ter o controle do inventário de sua empresa, principalmente para reduzir erros de contagem de produtos e garantir a sua disponibilidade para o cliente.

Porém, mesmo sendo de grande importância para as empresas, essa tarefa que envolve basicamente a contagem dos produtos em estoque, demanda grande quantidade de tempo, esforços por parte dos colaboradores e risco à segurança destes, além de gerar altos custos para a empresa, como explicitado por (WAWRLA; MAGHAZEI; NETLAND, 2019).

Para solucionar esses tipos de problemas, A WorkMarket (2020) apresenta que mais da metade dos empresários vem mostrando grande interesse na adoção de automação em suas empresas. Investimento que pode gerar uma economia de 73 bilhões de reais de acordo com Guto Ferreira, presidente da ABDI (GAZETA DO POVO, 2018).

Exemplos como a Vale, conseguiram economizar 5 milhões de dólares apenas utilizando sensores para identificação de problemas com os pneus de seus caminhões (GAZETA DO POVO, 2018). Ou a Realta Alimentos que conseguiu, além do aumento de sua produção em 52,78%, também reduzir o custo com mão de obra automatizando seus processos (MARAFON, et al, 2018).

Com o controle de estoques, que de acordo com Companik, et al (2018), é responsável por 30% do custo total em logística nos Estados Unidos, e, aliado ao conceito de indústria 4.0 que visa a automatização dos processos das indústrias tradicionais, surge uma ferramenta útil para a realização desta tarefa, os chamados drones.

Esses veículos remotamente controlados, vem ganhando grande espaço no mercado, sendo utilizados em vários segmentos como: monitoramento, vigilância, mapeamento, entre outros, gerará um valor de

mercado de 45,8 bilhões de dólares americanos no ano de 2025. (Market and Markets, 2019).

Consequentemente, muitas empresas como DSV (DSV, 2020), DroneScan (COMPANIK, 2018), criaram soluções para otimizar processos industriais como o controle de estoques. Utilizando principalmente drones capazes de ler algum tipo de código estampado, tais como QR Code e códigos de barra.

Entretanto, este trabalho tem como objetivo apresentar uma alternativa para empresas que não utilizam dessas ferramentas para identificar suas mercadorias.

2 REVISÃO BIBLIOGRÁFICA

Esta pesquisa tem como finalidade a elaboração de um sistema capaz de realizar o processo de inventário de estoques utilizando tecnologias de reconhecimento de imagens em ambientes naturais, como o armazém industrial.

2.1 Gerenciamento de Estoques

O setor logístico é extremamente importante para o sistema econômico de maneira geral, sendo responsável por proporcionar os bens e serviços para os consumidores ao mesmo tempo que assegura o escoamento da produção nas organizações (MOURA, 2006).

O conceito de logística abrange várias áreas do setor industrial como a de transportes, armazenagem e a gestão de estoques, sendo essa de grande relevância pois é ela que envolve grande parte do custo do produto, impactando no resultado da empresa. Por essa importância, a gestão de estoques é responsável por cerca de 40% a 50% do custo da logística, isso somado aos fundamentos da transformação digital, vem demandando grande investimento tecnológico para reduzir esses custos (AIRES; et al, 2019).

Na tentativa de reduzir esses custos, este documento apresenta uma solução para melhorar a eficiência no mapeamento de estoque, também conhecido como inventário.

2.2 Controle de Inventário

O inventário é fundamental para garantir a confiabilidade do estoque da empresa, garantindo que a quantidade armazenada é condizente com a informação do sistema virtual (SANTOS, 2013).

Essa tarefa envolve a listagem de todos os produtos em estoque, porém, mesmo não sendo muito complexa, pode despende muito tempo e dinheiro dependendo do tamanho da empresa (INVESTOR CONSULTING PARTNERS, 2020).

Frequentemente, o processo consiste em grupos de colaboradores da empresa que varrem todo o local de armazenagem identificando e contabilizando os itens a partir de algum sistema de caracterização que organizam os produtos pelo seu tamanho, cor ou departamento, podendo levar alguns dias para ser finalizado (CLIMBA, 2019).

Além do tempo demandado, outro fator que acarreta perdas para a empresa é a falta de precisão durante toda a operação, pois de acordo Wijffels et al (2016 apud Delaunay et al, 2007), esse tipo de erro gera um prejuízo de 2 milhões de dólares para cada 1 bilhão de dólares de bens vendidos.

Com o objetivo de apresentar uma solução para reduzir o tempo de duração e os erros relacionados ao processo de contagem, este trabalho promove uma alternativa mais barata das soluções já consolidadas que de acordo com Wijffels (2016), o custo é a principal barreira para a adoção desse tipo de solução.

2.3 Tecnologias de Reconhecimento de Imagens

Os sistemas de reconhecimento de imagens vem sendo alvo de grande procura principalmente na área científica e de engenharia. Uma das razões é

a variedade de possíveis aplicações do sistema em campos como: militar, biomedicina, manufatura, sistemas de segurança, entre outros (JAVIDI, 2001).

Essa tecnologia busca replicar a capacidade da visão humana em uma máquina, para assim analisar e processar imagens de interesse ajudando na redução das cargas de trabalho e de possíveis erros por parte dos colaboradores (JAVIDI, 2001).

Atualmente, pode-se observar exemplos desta multidisciplinaridade em trabalhos como o de Ruiz, et al (2021) que utiliza um algoritmo para executar a identificação e contagem de tarugos de aço, trazendo como resultado uma eficácia de aproximadamente 99%.

Outro exemplo a ser considerado é o trabalho de Nakajima, et al (2000) que utiliza o programa para reconhecer pessoas e estimar sua localização em tempo real, podendo ser utilizado para sistemas de vigilância.

2.4 Biblioteca OpenCV

Dentro da visão computacional, muitas tecnologias vêm sendo desenvolvidas para satisfazer a crescente demanda por soluções que permitam a interpretação do ambiente em que a máquina está inserida.

A princípio esse conceito era utilizado sob muitas restrições como um plano de fundo limpo, fontes padronizadas e boa iluminação, para que o computador tivesse melhores condições de analisar as imagens necessárias (GOSSELIN e THILLOU, 2007), como está mostrado na figura 2.

Figura 2 – Imagem em ambiente controlado



Fonte: <https://www.pyimagesearch.com/2017/01/02/rotate-images-correctly-with-opencv-and-python/>

Porém, para que essa tecnologia possa exercer todo o seu potencial no processo de automação, é necessário que ela seja capaz de exercer sua função em condições reais, como na indústria, por exemplo.

Com a implementação nesses locais, acarreta o surgimento de ruídos e condições aleatórias que prejudicam a análise como: iluminação, ângulo da câmera, distância do objeto, foco, entre outros (GOSELIN e THILLOU, 2007), como apresentado na figura 3.

Figura 3 – Imagem em ambiente natural

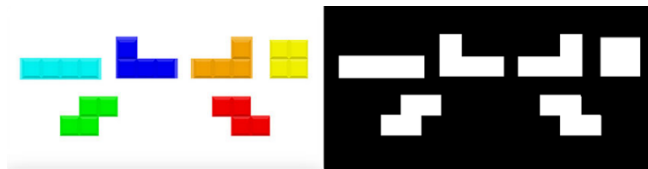


Fonte: GOSELIN E THILLOU (p. 308)

Para lidar com todas essas variabilidades foi desenvolvido no ano de 2000 pelo cientista Gary Bradsky, a biblioteca OpenCv que vem contribuindo para o desenvolvimento de soluções de visão computacional e inteligência artificial (OPENCV, 2021).

Com essa biblioteca várias etapas do processamento de imagens são facilitadas, como por exemplo, a extração de regiões de interesse para separar a área de análise do restante da imagem, a detecção de contornos, *Thresholding* para demarcar os limites dos objetos e colocá-los em destaque a partir de uma máscara como visto na figura 4.

Figura 4 – Imagem original junto com sua máscara



Fonte: <https://www.pyimagesearch.com/2018/07/19/opencv-tutorial-a-guide-to-learn-opencv/>

Com a máscara, será realizada a operação de *bitwise* para comparar com a imagem original e retirar as áreas indesejadas para o processo.

Neste trabalho, o uso dessa biblioteca será restrito a identificação de rótulos de produtos podendo ser códigos de barras ou qualquer forma de identificação textual, para serem processados e utilizados em outras tecnologias posteriores.

2.5 Reconhecimento de Textos

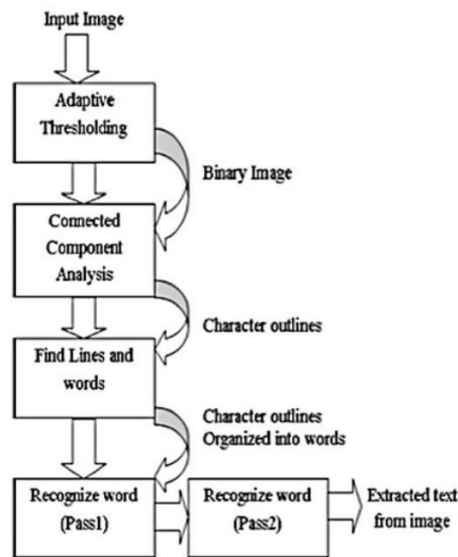
Após a extração e o processamento das imagens com o uso do OpenCV, obtém-se como resultado as regiões de interesse dos objetos analisados, ou seja, as suas áreas de texto a serem identificadas. Com isso, essas imagens podem ser utilizadas em tecnologias do tipo *Optical Character Recognition* também conhecido por OCR.

O OCR tem como finalidade converter o texto de imagens que podem estar escritas à mão ou impressa, em cadeias de caracteres que podem ser manipuladas por programas de computador (RICE et al, 1999).

Um dos exemplos mais famosos desse segmento é o Tesseract criado na década de 80 por Hewlett Packard e hoje é mantido pelo Google. Ele é capaz de reconhecer textos em mais de 100 línguas diferentes.

O Tesseract recebe a imagem processada pelo OpenCv e simplifica as palavras para apenas contornos para então realizar os processos de reconhecimento. Essa etapa é dividida em duas partes, na primeira, os textos são enviados a um algoritmo classificador que irá traduzi-los em classes conhecidas, ou seja, em caracteres comumente conhecidos como os da língua portuguesa por exemplo. A segunda etapa é encarregada de compreender o contexto daquilo caracterizado na primeira etapa (AUDICHYA e SAINI, 2017). Um fluxograma desse processo é apresentado na figura 5 de forma simplificada.

Figura 5 – Fluxograma do Tesseract



Fonte: A Study to Recognize Printed Gujarati Characters Using Tesseract OCR

Por fim, o resultado do processo pode ser armazenado e editado para diversas aplicações em que essa informação seja relevante.

3 METODOLOGIA

Na primeira etapa do projeto foi realizada uma pesquisa que abrange principalmente a área de gestão de estoques e de tecnologias para reconhecimento de padrões de imagem.

Na gestão o foco está no método de contagem de produtos em estoque e na área de reconhecimento de imagens, serão ressaltados os tópicos de processamento de imagens para reconhecimento das regiões textuais.

A pesquisa realizada no campo da visão computacional mostrou a possibilidade de realizar operações como a de inventário de forma mais eficaz através de ferramentas de reconhecimento óptico de caracteres já existentes. A escolhida foi o Tesseract.

Além da leitura dos textos de forma direta e fácil, essa tecnologia permite a extração das coordenadas dos caracteres encontrados, o que pode ser uma informação muito poderosa.

Testes diretos com essa tecnologia foram aplicados em algumas fotos tiradas em um armazém industrial. Os resultados mostraram a necessidade da realização de ajustes prévios para que o reconhecimento possa ser feito de maneira correta, para isso, a biblioteca OpenCv foi utilizada.

A utilização dessa biblioteca no projeto pode ser dividida em duas partes, sendo elas, a busca pela região de interesse e o tratamento de ruídos, que no projeto em questão, podem ser definidos como distorções visuais que atrapalham no bom funcionamento do Tesseract.

A região estabelecida como de interesse foi aquela contendo apenas os códigos que identificam os produtos que devem ser contabilizados.

Estudos mostraram que o tempo de paralização durante a contagem estão diretamente ligados a maiores custos, por tanto, para acelerar o processo de inventário, as imagens utilizadas foram de pallets contendo diversas caixas, o que gera necessidade de correta extração de múltiplas regiões de interesse por imagem.

Nessa etapa, utilizou-se o fundo dos rótulos presos as caixas como delimitador das regiões de interesse, por elas possuírem todas o mesmo padrão, foi possível através de funções do OpenCv, dar um destaque maior, permitindo a localização de seus contornos para que possam ser separadas do resto da imagem.

A remoção de ruídos foi aplicada diretamente nas regiões encontradas na etapa anterior, na tentativa de criar um destaque nos códigos contidos nos rótulos, permitindo que o produto seja identificado mesmo em imagens tiradas em ambientes com diferentes luminosidades, por exemplo.

Em todo o conjunto operado pelo OpenCv, algumas funções foram extremamente importantes durante todo o processamento, como:

- `cvtColor` que converte o sistema de cor da imagem para outro, como por exemplo transformar do sistema padrão da biblioteca, o BGR (blue, green, red), para a escala de cinza.
- `Thresholding` que geralmente é acionada para transformar a imagem em escala de cinza para uma binária, ou seja, segmentar

a imagem em apenas dois valores de intensidade dos pixels para diferenciar aqueles que são de interesse do resto, formando a chamada máscara.

- Bitwise que são operações aritméticas aplicadas em imagens, comparando bits (dígitos binários) de duas imagens do mesmo tamanho, sendo uma delas geralmente, uma máscara da original.
- Inrange que quando aplicada em uma imagem HSV, nos permite extrair todo o espaço da cor desejada através do uso de limitadores.

Em casos com mais de um código em uma mesma região do interesse, foi possível através do Tesseract, a obtenção das coordenadas de todos os caracteres identificados, permitindo que não haja repetições em cada leitura.

Após a criação dos algoritmos de reconhecimento, iniciou-se a procura pelo *hardware* que deveria fornecer a infraestrutura necessária para suportar os programas instalados e permitir se conectar com outros dispositivos via Wifi ou diferentes protocolos de comunicação como LoraWan ou SigFox, por exemplo.

A partir da busca, a opção que mais se destacou foi o Raspberry Pi, *um single board computer* (SBC) de pequeno porte que pode ser facilmente preso a um drone para se locomover pelo ambiente. Ele foi criado com intuito de ser um ambiente de programação de baixo custo focado na linguagem Python utilizada nos softwares de reconhecimento de imagens.

O Raspberry Pi também possui entradas do tipo CDI que possibilitaram o acoplamento da câmera responsável pela captação das imagens dentro do armazém.

A cada código identificado no processo, o Raspberry permitiu, através de sua comunicação por Wifi, o envio das informações para o Azure, um servidor em nuvem por meio de uma API desenvolvida.

Os dados foram armazenados em um banco de dados, podendo ser visualizados em softwares de inteligência de negócios, como o Power BI da Microsoft.

Por fim, foi elaborada uma conclusão a partir dos resultados obtidos nos testes práticos, avaliando sua performance a fim de determinar sua viabilidade e pontos de melhoria.

4 RESULTADOS E DISCUSSÃO

Este item apresenta os resultados do trabalho e está dividido em 5 subcapítulos que englobam o passo a passo detalhado do processo, desde a captura da imagem pelo operador até o envio dos dados ao servidor.

4.1 MODELOS DE RÓTULOS PARA PROTOTIPAÇÃO

Em galpões industriais é comum fixarem os rótulos diretamente nas caixas para facilitar a identificação e localização dos produtos. Para sua própria organização, cada empresa costuma utilizar um padrão de códigos e rótulos.

Como padrão para o desenvolvimento do protótipo, foi estabelecido códigos de 10 caracteres impressos em folha de cor branca e fonte preta.

A figura 6 representa uma foto de um pallet padrão contendo 6 rótulos com os respectivos códigos e nomes do produto de cada caixa, capturada por uma câmera acoplada a um drone pilotado por um operador capacitado.

Figura 6 – Rótulos utilizados como modelo



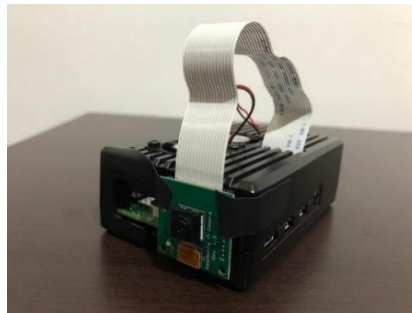
Fonte: Os autores (2021)

4.2 SISTEMA DE CAPTURA DE FOTOS

As caixas dos produtos nos pallets são fotografadas pelo operador do drone, através do módulo de câmera para Raspberry Pi com 5 megapixels.

Na figura 7 mostra a câmera utilizada nesse trabalho, acoplada ao Raspberry através de sua porta CDi.

Figura 7 – Câmera acoplada ao RaspberryPi



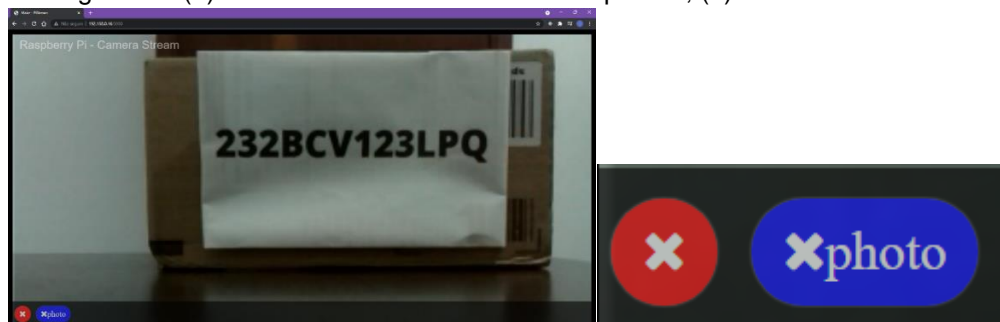
Fonte: Os Autores (2021)

O acesso à essa câmera é feito de forma remota através de um servidor local gerado no Raspberry, possibilitando a transmissão em tempo real para outros computadores na mesma rede de internet.

A figura 8 (a) representa a interface onde o usuário tem a visão do drone quando posicionado para capturar a imagem da caixa.

O sistema de transmissão possui dois botões de ação, como ilustrado na figura 8 (b), sendo o vermelho responsável pela finalização da transmissão, e o azul pela captura da imagem e envio para tratamento das regiões de interesse.

Figura 8 – (a) Sistema de transmissão em tempo real; (b) Botões do sistema



(a)

(b)

Fonte: Os Autores (2021)

4.3 CAPTURA DAS REGIÕES DE INTERESSE

4.3.1 REGIÕES DE INTERESSE PRELIMINARES

No processo de captura das regiões de interesse, o objetivo é a obtenção de uma nova foto para cada rótulo do pallet sendo analisado.

Primeiramente, as imagens capturadas são fixadas a um tamanho fixo de 800 pixels de altura e comprimento proporcional a razão de suas dimensões originais.

Após a formatação, a imagem é transformada para uma escala de cinza. A figura 9 (a) mostra o resultado do processo com apenas um canal de cor que irá de 0 (preto) a 255 (branco).

Além disso, foi aplicado o *Gaussian blur* e uma transformação morfológica chamada *erosion* para reduzir ruídos e detalhes indesejados da imagem, como ilustrado na figura 9 (b).

Figura 9 – (a) Escala de cinza; (b) Gaussian blur e erosion

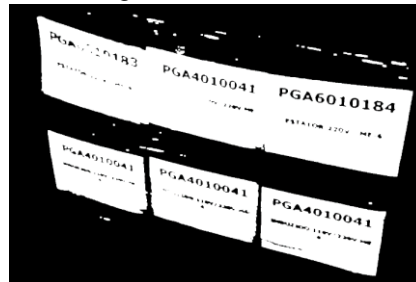


(a) (b)
Fonte: Os Autores (2021)

Inicia-se a partir disso, a etapa de *threshold* que é uma etapa intermediária no processamento que resume a coloração das imagens em apenas dois valores, 0 ou 255, a partir de um valor fixado.

Nos testes realizados, esse valor fixado foi 180. A figura 10 apresenta como todo pixel que possuir um valor abaixo do estabelecido, se tornará 0 e acima, será 255, formando o que é comumente conhecido como máscara.

Figura 10 – Máscara



Fonte: Os Autores (2021)

Como os rótulos utilizados nos testes possuem fundo branco, conforme mostrado na figura 10, a máscara realizada na etapa anterior contribui para a captura das suas coordenadas aproximadas através de funções no programa que analisam os contornos destacados em branco.

Ao fim desse primeiro processo, a região destacada pelas coordenadas é separada da imagem original e armazenada em memória. Porém, apesar da captura região de interesse aproximada, ela pode vir acompanhada de outras sem as informações desejadas.

4.3.2 TRATAMENTO DAS REGIÕES PRELIMINARES

Após a análise de várias possíveis regiões de interesse, foi constatado que as regiões que possuem os códigos desejados possuem uma área não inferior a 10000 pixels. A partir disso, a primeira etapa é a filtragem das áreas com valor maior que o mencionado.

Na figura 11, tem-se a direita uma das regiões que não possui nenhum tipo de informação e à esquerda, uma das que possuem o rótulo desejado, porém ainda com uma angulação e com partes indesejadas, como a caixa do produto onde o rótulo está fixado. Por isso a necessidade de uma filtragem inicial e tratamento posterior.

Figura 11 – Exemplo de ROI preliminares



Fonte: Os Autores (2021)

As imagens aceitas, como aquela à esquerda na figura 11, passam por uma operação chamada de *bitwise not* onde os valores dos pixels são invertidos, ou seja, todos os seus bits são trocados, transformando aqueles que tinham o valor 0 em 1 e vice-versa.

Na figura 12 (a), pode-se observar o resultado da operação de *bitwise not* onde a cor do fundo rótulo se tornou preta, destacando essa região.

Através da operação de *bitwise not*, pode-se utilizar a operação do OpenCV chamada de *InRange*, que a partir da escolha dos limites da cor preta no espaço de cor HSV, permite a formação de uma nova máscara binária, como mostrado na figura 12 (b).

Figura 12 – (a) Bitwise not; (b) Nova Máscara Binária



Fonte: Os Autores (2021)

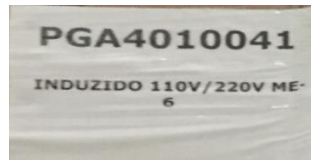
O padrão HSV é uma abreviatura para os termos: *hue* que é responsável pela tonalidade das cores, *saturation*, responsável pela pureza da imagem e *value* que define o brilho da cor.

Nessa etapa do processo, repete-se a função para localizar os contornos da região de interesse. Os contornos fechados passam por uma aproximação de figura geométrica, essa etapa é realizada pela função da biblioteca OpenCV *approxPolyDP*.

Caso a figura geométrica possua 4 vértices, é assumido que o rótulo foi encontrado pois, nos testes realizados, eles possuem todos forma retangular.

Por fim, é aplicada uma transformação de perspectiva utilizando os vértices como referência para ter uma visão frontal da região encontrada, como na figura 13.

Figura 13 – Região de interesse final



Fonte: Os Autores (2021)

4.4 RECONHECIMENTO DOS CARACTERES NAS REGIÕES

4.4.1 PRIMEIRA TENTATIVA DE RECONHECIMENTO

Logo após o término da captura das regiões de interesse, ocorre uma tentativa direta do reconhecimento dos caracteres contidos na imagem através de uma ferramenta de OCR chamada de Tesseract.

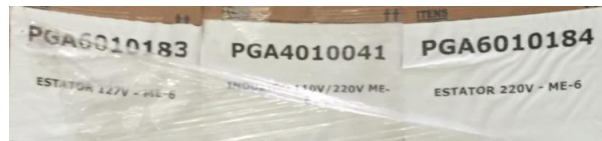
Foi utilizado nesse trabalho o pacote Pytesseract pertencente a essa ferramenta de reconhecimento de caracteres do Google, principalmente pela sua praticidade de execução e por permitir fácil acesso ao motor do OCR através da linguagem de programação Python, utilizada no projeto.

Nesta tentativa, caso o OCR reconheça alguma combinação de caracteres com a quantidade padrão estabelecida pela empresa, ele irá averiguar a existência desse código no banco de dados da empresa.

Nos testes realizados, estabeleceu-se uma quantidade de 10 caracteres como padrão de código desejado e a utilização de um arquivo csv simulando a consulta no banco de dados.

Em algumas situações, como no caso da figura 14, é possível que a região de interesse contenha mais de um rótulo. Isso pode ocorrer pela angulação da foto, luminosidade, proximidade dos rótulos, entre outros.

Figura 14 – ROI com múltiplos rótulos



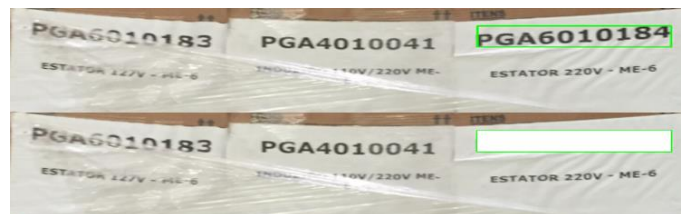
Fonte: Os Autores (2021)

Após a confirmação da existência do código encontrado no banco de dados, deve-se verificar se há outros rótulos além desse. Para isso, são capturadas as coordenadas cartesianas do ponto superior à esquerda do início do código, a sua altura e largura.

Com essas informações adquiridas, é possível realizar o cálculo da área pertencente ao código e descobrir o quanto ela ocupa da imagem total. Após vários testes, admitimos que caso a área ocupe menos que 10% da área total da região de interesse, possui uma alta probabilidade de existir outros rótulos naquela imagem.

A figura 15 demonstra uma região de interesse com mais de um rótulo, nessa situação um retângulo é desenhado a partir das dimensões adquiridas e preenchido com branco, evitando que o mesmo código seja lido mais de uma vez após futuro tratamento.

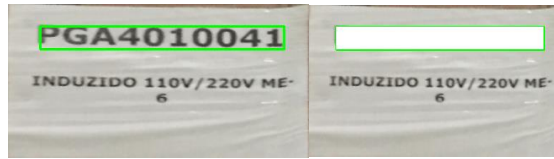
Figura 15 – Reconhecimento em múltiplos rótulos



Fonte: Os Autores (2021)

A figura 16 representa uma região de interesse de rótulo único com área maior que 10% da imagem total. Nesse caso, é assumido que todos os rótulos presentes foram encontrados.

Figura 16 – Reconhecimento em rótulo único



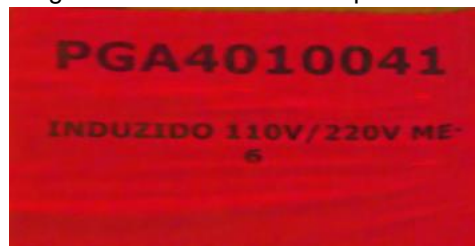
Fonte: Os Autores (2021)

4.4.2 TRATAMENTO EM CASOS DE NÃO IDENTIFICAÇÃO

Em situações em que não foi possível o reconhecimento de forma imediata, a região de interesse é enviada para um novo tratamento.

Esse tratamento envolve as mesmas operações já realizadas na etapa tratamento das regiões de interesse preliminares, começando pela conversão da imagem para o padrão HSV representada na figura 17.

Figura 17 – ROI convertida para HSV



Fonte: Os Autores (2021)

Novamente, a máscara binária é formada a partir da função *InRange* com o objetivo de destacar o código em preto da figura 17. Porém, neste caso o limite superior não é fixado.

A razão dessa flexibilidade decorre do fato das fotos serem tiradas em ambientes naturais, ou seja, onde não há o controle da luminosidade, angulação da caixa, fontes dos rótulos, reflexos, entre outros.

Essa falta de controle faz com que cada região de interesse necessite de uma intensidade de destaque diferente para que o Tesseract possa interpretar o rótulo corretamente.

Para atingir o nível de destaque suficiente para a leitura, altera-se apenas a intensidade de brilho para que o espaço de cor preta continue sendo totalmente contemplado, sendo que o brilho pode variar de 0 a 255.

Após experimentos, foi obtido melhores resultados estabelecendo um intervalo de 90 a 150, além de um passo de 3 para otimização do processo.

Na figura 18, é possível notar a diferença do realce do texto à esquerda com o brilho em 90, para o da direita com 130.

Figura 18 – Máscara com diferentes intensidades de brilho

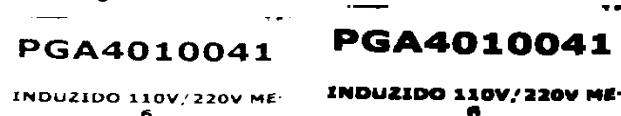


Fonte: Os Autores (2021)

A máscara sofre uma nova transformação morfológica conhecida como *dilate*, que é exatamente o oposto da já utilizada *erosion*, mas nesse caso, ela aumenta os ruídos brancos da imagem. Em seguida, tem seus pixels invertidos através da operação de *bitwise not*.

A imagem 19 mostra a figura com a diferença da região de interesse final sem e com a aplicação morfológica, respectivamente.

Figura 19 – Bitwise not sem e com dilate



Fonte: Os Autores (2021)

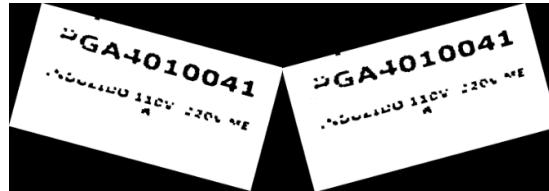
Nesse momento, há a tentativa de leitura pelo Tesseract como descrito anteriormente.

Em algumas situações, o rótulo ou o próprio código podem ter algum grau de angulação, o que pode acarretar um erro de leitura mesmo que a intensidade de brilho esteja correta.

Nesses casos, após a falha na identificação do código, o programa rotaciona a imagem primeiramente, no sentido horário e executa uma nova leitura. Caso não tenha sucesso, a imagem é novamente rotacionada, porém, em sentido anti-horário.

A imagem 20 ilustra a rotação das regiões de interesse em ambos os sentidos.

Figura 20 – Rotação horária e anti-horária



Fonte: Os Autores

Se após o processo, o Tesseract não conseguir realizar a leitura da imagem, a intensidade do brilho é aumentada em 3 unidades e o processo é reiniciado.

A figura 21 ilustra uma região de interesse onde o reflexo do plástico impediu a leitura de um dos códigos pelo Tesseract.

Figura 21 – Exemplo de erro na identificação



Fonte: Os Autores (2021)

Em casos como da figura 21 em que o valor atinja o máximo de 150 sem que o código seja identificado, a região é salva em uma pasta para futura análise.

4.5 ENVIO DOS DADOS AO SERVIDOR

Com o sucesso na leitura do código contido na região de interesse, os dados do produto e sua quantidade são agrupados para o formato json. Esse formato possibilita o envio das informações, por meio da internet, a um servidor de banco de dados remoto, na nuvem.

Para realizar a operação, é utilizada uma API, ou seja, uma interface de programação de aplicativos que permite a interação entre diferentes aplicações, tornando mais fácil o uso e a manipulação dos dados entre elas.

A API desenvolvida para esse projeto foi também escrita em linguagem Python, utilizando principalmente o framework Flask que facilita a construção dessas interfaces, oferecendo funções preestabelecidas para agilizar e simplificar o processo de desenvolvimento.

Os dados são enviados à API por meio da arquitetura *REST* que define alguns conjuntos de operações, permitindo a comunicação entre sistemas da internet.

A operação utilizada para o envio é chamada de PUT ou Update, ela é responsável também pela atualização daquele dado, caso ele já exista no banco de dados.

Ao final da operação, as informações são adicionadas a um banco de dados hospedado na nuvem para consultas futuras. O banco utilizado nesse projeto foi o CosmosDB da Microsoft.

A quantidade de cada modelo pode então ser observada em um painel gráfico, permitindo ao usuário realizar diversos tipos de análises.

5 CONSIDERAÇÕES FINAIS

Através do uso de visão computacional, foi possível o desenvolvimento de um sistema para reconhecimento de códigos em rótulos de produtos em armazéns industriais.

O sistema criado é composto por um microprocessador Raspberry Pi com o seu módulo de câmera acoplado para captura e processamento das imagens e com uma capa com ventoinhas para dissipação do calor gerado, um drone comercial Phantom 4 Pro apenas para a locomoção e uma bateria de 10000 mAh para alimentação da CPU.

O desenvolvimento do projeto passou por algumas dificuldades pela falta de meios de comunicação e acesso às mídias capturadas pela câmera do drone fornecido, pois não foi possível acessá-las através de cabos durante o voo, ou por vias remotas, utilizando o sistema desenvolvido para se comunicar com o drone pela internet.

Para contornar o problema, foi utilizado um módulo de câmera próprio para o Raspberry Pi, que por sua vez, vai fixado junto ao drone para capturar as imagens e realizar a leitura dos códigos, deixando o drone como apenas um veículo de locomoção.

Para que o inventário seja bem-sucedido utilizando o projeto apresentado, algumas considerações em relação a estrutura do armazém são necessárias. Em casos em que o mesmo pallet abriga diferentes produtos, todas as caixas devem conter o rótulo voltado para fora, além de existir espaço suficiente para que o drone possa se movimentar em torno do pallet.

Em contrapartida, a hipótese proposta não engloba como solução para a caixa central, isto é, aquela caixa rodeada em todos os lados por outras caixas, impedindo que o rótulo fique visível.

Quando o pallet contém apenas um rótulo para todo o pallet, é entendido que a empresa abrigue uma quantidade pré-estabelecida de caixas, necessitando apenas a identificação do produto, ou que exista o mesmo código em todas as caixas.

Testes a respeito da contagem de caixas através do cálculo da sua área por uma imagem, se mostraram ineficazes em ambientes onde não há a capacidade de proporcionar uma medida como referência em cada foto tirada, sendo necessário um estudo maior sobre o tema.

Falhas na identificação dos códigos foram encontradas em pallets fechados onde o plástico filme transparente que une as caixas envolve os códigos.

Apesar das dificuldades, foi possível desenvolver o projeto atendendo os objetivos de pesquisa inicialmente definidos. As falhas e descobertas obtidas poderão servir de base para estudos mais aprofundados posteriores.

REFERÊNCIAS

AIRES, C.S.F; ALMEIDA, G.J.; SILVEIRA, S.O.; INTELIGÊNCIA ARTIFICIAL NA GESTÃO DE ESTOQUE. **X FATECLOG**, Guarulhos, p. 1-7, 2019. Disponível em: <https://fateclog.com.br/anais/2019/INTELIG%c3%8aNANCIA%20ARTIFICIAL%20NA%20GEST%c3%83O%20DE%20ESTOQUE.pdf>. Acesso em: 05 abr. 2021.

ARYAL, S. Integrating camera recognition and RFID system for assets tracking and warehouse management. **Central Ostrobothnia University of Applied Sciences**, Ylivieska, p. 1-36, 2012. Disponível em: <https://www.theseus.fi/handle/10024/40540>. Acesso em: 05 abr. 2021.

AUDICHYA, M. K.; SAINI, J. R. A Study to Recognize Printed Gujarati Characters Using Tesseract OCR. **Engineering, Technology and Applied Science Research**, Ahmedabad, v. 5, n. 9, p. 1505-1510, 2017. DOI: 10.22214/ijraset.2017.9219. Disponível em: https://www.researchgate.net/profile/Milind-Audichya/publication/320167029_A_Study_to_Recognize_Printed_Gujarati_Characters_Using_Tesseract_OCR/links/59d22de9a6fdcc181ad5f09a/A-Study-to-Recognize-Printed-Gujarati-Characters-Using-Tesseract-OCR.pdf. Acesso em: 02 mai. 2021.

BAGDANOV, A. D.; RICE, S. V.; NARTKER, T. A. The OCR Frontiers Toolkit, Las Vegas, 1999. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.216.9106&rep=rep1&type=pdf>. Acesso em: 10 mai. 2021

CASTRO, S. A.; SILVA, A. C. DAS N.; BROTTI, K. C. DAS N. S. A inteligência artificial e os sistemas especialistas aplicados à produção. **Revista Processando o Saber**, v. 6, p. 27-46, 1 out. 2014. Disponível em: <https://www.fatecpg.edu.br/revista/index.php/ps/article/view/81>. Acesso em: 05 abr. 2021.

COMPANIK, E.; GRAVIER, M.J.; FARRIS II, M.T. Feasibility of warehouse drone adoption and implementation. **Journal of Transportation Management**, Michigan, v. 28, n. 2, p. 30-48, 2018. DOI: 10.22237/jotm/1541030640. Disponível em: <https://digitalcommons.wayne.edu/jotm/vol28/iss2/5/>. Acesso em: 07 abr. 2021.

COVID-19 Impact on Logistics & Supply Chain Industry Market by Industry Verticals (Automotive, FMCG, Healthcare, Energy & Utilities, Industrial Machinery & Equipment), Mode of Transport (Roadways, Railways, Airways, Maritime), Region - Global Forecast to 2021. **Markets and Markets**, Illinois, abr. de 2020. Disponível em: <<https://www.marketsandmarkets.com/Market->

Reports/covid-19-impact-on-logistics-supply-chain-industry-market-244593137.html>. Acesso em: 25 de mar. de 2021.

DSV improves warehouse operations with drone system. **DSV Global Transport and Logistics**, 2021. Disponível em: <<https://www.dsv.com/en/about-dsv/press/news/com/2020/11/dsv-improves-warehouse-operations-with-drone-system>>. Acesso em: 10 mai. 2021

HERNÁNDEZ-RUIZ, A. C.; MARTINEZ-NIETO, J. A.; BULDAIN-PÉREZ, J. D. Steel Bar Counting from Images with Machine Learning. **Machine Learning and Smart Sensing**, Zaragoza, 2021. DOI: <https://doi.org/10.3390/electronics10040402>. Disponível em: <https://www.mdpi.com/2079-9292/10/4/402>. Acesso em: 06 mai. 2021.

HONG-YING, S. The Application of Barcode Technology in Logistics and Warehouse Management. **2009 First International Workshop on Education Technology and Computer Science**, Wuhan, p. 732-735, 2009. DOI: 10.1109/ETCS.2009.698. Disponível em: <https://ieeexplore.ieee.org/abstract/document/4959416>. Acesso em: 05 abr. 2021.

Introduction to OpenCV-Python Tutorials. **OpenCV Open Source Computer Vision**, 2021. Disponível em: <https://docs.opencv.org/master/d0/de3/tutorial_py_intro.html>. Acesso em: 16 de mai. de 2021. INVENTÁRIO de estoque: o que é e como fazer?. **Climba Commerce**, 2019. Disponível em: <<https://www.climba.com.br/blog/inventario-de-estoque-o-que-e-e-como-fazer/>>. Acesso em: 15 mai. 2021.

INVENTÁRIO DE ESTOQUE: O QUE É, PARA QUE SERVE E COMO FAZER. **Investor Consulting Partners**, 2020. Disponível em: <<https://investorcp.com/gestao-ativo-imobilizado/o-que-e-inventario-estoque/>>. Acesso em: 14 mai. 2021

JAVIDI, B. **Image Recognition and Classification**. Nova York: Taylor & Francis, 2005.

LOGISTICS Automation Market by Component, Function (Warehouse and Storage Management, Transportation Management), Organization Size (SMES, Large Enterprises), Vertical (3PL, Retail and Ecommerce, FMCG, Manufacturing), and Region - Global Forecast to 2026. **Markets and Markets**, Illinois, mar. de 2021. Disponível em: <<https://www.marketsandmarkets.com/Market-Reports/logistics-automation-market-83657139.html>>. Acesso em: 25 de mar. de 2021.

MARAFON, C. et al. BENEFÍCIOS DO INVESTIMENTO EM AUTOMAÇÃO NO PROCESSO DE EMPACOTAMENTO DE FARINHA DE TRIGO. **UCEFF**, Itapiranga, p. 72-87, 2018.

MOURA, B. Enquadramento geral. *In*: MOURA, B. **Logística: Conceitos e Tendências**. Lisboa: Centro Atlântico, 2006. p. 15-49.

NAKAJIMA, C.; PONTIL, M.; POGGIO, T. People recognition and pose estimation in image sequences. **Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium**, Como, 2000. DOI: 10.1109/IJCNN.2000.860771. Disponível em: <https://ieeexplore.ieee.org/abstract/document/860771/authors#authors>. Acesso em: 07 mai. 2021

OLIVEIRA, A.A.; LOPE, J.L.; BURI, M.R. GESTÃO DE ESTOQUE: AS DIFICULDADES DE INVENTÁRIO EM UMA EMPRESA DE LOGÍSTICA LOCALIZADA EM BARUERI – SP. **Revista da Faculdade Eça de Queiros**, São Paulo, n. 4, p. 1-15, 2011.

PARA garantir competitividade, empresas no Brasil se adaptam à indústria 4.0. **Gazeta do Povo**, São Paulo, 27 mai. 2021. Disponível em: <https://www.gazetadopovo.com.br/economia/nova-economia/para-garantir-competitividade-empresas-no-brasil-se-adaptam-a-industria-40-1fdb7cnqjak4gh98hofy8k6ut/>. Acesso em: 07 abr. 2021.

PIHIR, I.; TOMIČIĆ-PUPEK, K.; FURJAN, M.T.; Digital Transformation Insights and Trends. **Central European Conference on Information and Intelligent Systems**, Varaždin, p. 141-149, 2018. Disponível em: <http://archive.ceciis.foi.hr/app/public/conferences/2018/Proceedings/ETICT/ETICT-6.pdf>. Acesso em: 28 mar.2021.

SACOMANO, J.B.; SÁTYRO, W.C. Indústria 4.0: Conceitos e elementos formadores. *In*: SACOMANO, J.B. et al. (org.). **Industria 4.0: conceitos e fundamentos**. São Paulo: Blucher, 2018. p. 27-44.

SANTOS, D.C.I. GESTÃO DE ESTOQUE. **FEMA**, Assis, p. 11-44, 2013.

SOBRE Watson IoT Platform. **IBM**, 2021. Disponível em: https://cloud.ibm.com/docs/IoT?topic=IoT-about_iotplatform>. Acesso em: 24 mai. 2021

THILLOU, C.; GOSSELIN, B. Natural Scene Text Understanding. **Vision Systems: Segmentation and Pattern Recognition**, Mons, p. 307-332, 2007. DOI: 10.5772/4966. Disponível em: https://www.researchgate.net/publication/221786127_Natural_Scene_Text_Understanding. Acesso em: 04 mai. 2021

THINKGSPEAK. **ThinkSpeak**, c2021. Página Inicial. Disponível em: <https://thingspeak.com/>>. Acesso em: 20 de mai. de 2020.

UNMANNED Aerial Vehicle (UAV) Market by Vertical, Class, System, Industry (Defense & Security, Agriculture, Construction & Mining, Media & Entertainment), Type, Mode of Operation, Range, Point of Sale, MTOW and

Region - Global Forecast to 2025. **Markets and Markets**, Illinois, out. de 2019. Disponível em: < <https://www.marketsandmarkets.com/Market-Reports/unmanned-aerial-vehicles-uav-market-662.html>>. Acesso em: 10 de abr. de 2021.

WAWRLA, L.; MAGHAZEI, O.; NETLAND, T.; Applications of drones in warehouse operations. **Chair of Production and Operations Management Department of Management, Technology and Economics**, Zurich, p. 1-12, 2019. Disponível em: https://ethz.ch/content/dam/ethz/special-interest/mtec/pom-dam/documents/Drones%20in%20warehouse%20opeations_POM%20white%20paper%202019_Final.pdf. Acesso em: 05 abr. 2021.

WIFFELS, L. *et al.* An enhanced cycle counting approach utilising historical inventory data. **8th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2016: Troyes, France, 28—30 June 2016**, Troyes, v. 49, n. 12, p. 1347-1352. DOI: <https://doi.org/10.1016/j.ifacol.2016.07.748>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2405896316310291>. Acesso em: 06 mai. 2021.

WORKMARKET 2020 In (Sight) Report. **Work Market**, Nova Jersey, 2021. Disponível em: <https://www.workmarket.com/go/2020-insight-report-what-ai-automation-mean-for-work>. Acesso em: 05 abr. 2021.