

CENTRO UNIVERSITÁRIO CARIOCA

Ezequiel Nascimento Campos

EXPLORANDO A VISÃO COMPUTACIONAL COM ANÁLISES DE
IMAGENS UTILIZANDO AWS REKOGNITION

Rio de Janeiro
2023

EZEQUIEL NASCIMENTO CAMPOS

EXPLORANDO A VISÃO COMPUTACIONAL COM ANÁLISE DE
IMAGENS UTILIZANDO AWS REKOGNITION

Trabalho de Conclusão de Curso
apresentado ao Centro
Universitário Carioca, como
requisito parcial à obtenção do
grau de Bacharel em Ciência da
Computação.

Orientadora: Prof.^a Daisy Cristine Albuquerque da Silva, M.Sc.

**Rio de Janeiro
2023**

C172 Campos, Ezequiel Nascimento.

Explorando a visão computacional com análises de imagens utilizando AWS Rekognition. / Ezequiel Nascimento Campos. Rio de Janeiro, 2023.

70f.

Orientadora: Daisy Cristine Albuquerque da Silva.

Trabalho de Conclusão de Curso (graduação em Ciência da Computação) Centro Universitário UniCarioca - Rio de Janeiro, 2023.

1. Visão computacional. 2. Inteligência artificial 3. Computação em nuvens. I. Silva, Daisy Cristine Albuquerque da, prof. orient. II. Título.

CDD 005

EXPLORANDO A VISÃO COMPUTACIONAL COM ANÁLISE DE IMAGEM UTILIZANDO AWS REKOGNITION

Trabalho de Conclusão de Curso
apresentado ao Centro
Universitário Carioca, como
requisito parcial à obtenção do
grau de Bacharel em Ciência da
Computação.

Banca Examinadora

Prof.^a Daisy Cristine Albuquerque da Silva, M.Sc. – Orientadora

Centro Universitário Carioca

Prof.^o Sergio Assunção Monteiro - M.Sc. – Professos Convidado

Centro Universitário Carioca

Prof.^o André Luiz Avelino Sobral - M.Sc. – Coordenador

Centro Universitário Carioca

Agradecimentos

Gostaria de agradecer à minha família, que sempre estiveram presentes e me apoiaram incondicionalmente em todas as etapas da minha vida acadêmica. Sem o amor, incentivo e suporte de vocês, eu não teria chegado até aqui. Vocês são minha fonte de inspiração e força, e sou imensamente grato por tudo o que fizeram por mim. A minha professora orientadora Daisy Albuquerque que aceitou me orientar no desenvolvimento desse TCC e que tem sido uma guia excepcional desde o ano passado. Sua expertise, paciência e dedicação em me auxiliar neste TCC foram fundamentais para o seu desenvolvimento e conclusão. Não posso deixar de agradecer aos amigos que estiveram ao meu lado principalmente a minha amiga, Raquel, seja compartilhando experiências, trocando ideias ou oferecendo suporte emocional. Seus conselhos, encorajamento e amizade foram fundamentais para tornar essa jornada mais leve e significativa. Gostaria de agradecer também a minha equipe do meu trabalho, no IBRE FGV, com eles aprendi muito sobre Python, IA e ML. Meu muito obrigado a cada um de vocês!

Resumo

Este trabalho teve como objetivo principal explorar as capacidades da plataforma Amazon Web Services e sua ferramenta AWS Rekognition para aplicar técnicas de visão computacional e inteligência artificial na análise de imagens. O estudo foi motivado pela curiosidade em compreender o funcionamento das câmeras de dispositivos móveis, especialmente devido ao aumento do uso de inteligência artificial e visão computacional nesses dispositivos nos últimos anos.

Por meio do uso de algoritmos de aprendizado de máquina, é possível extrair resultados impressionantes a partir de hardware limitado, otimizando recursos e executando tarefas complexas em tempo real. A identificação e reconhecimento de objetos, rostos e cenários em tempo real abrem caminho para melhorias na qualidade das fotografias e a criação de experiências interativas para os usuários.

Além disso, a utilização da computação em nuvem tornou-se uma parte fundamental deste projeto, permitindo economizar tempo e recursos computacionais locais. Ao combinar Python, computação em nuvem e inteligência artificial, foi possível implementar algoritmos de visão computacional de forma eficiente, aproveitando a escalabilidade e poder de processamento oferecidos pela AWS.

Palavras-chaves: *Machine Learning*, Visão computacional, Inteligência artificial, Computação em nuvem

Abstract

This work aimed to explore the capabilities of the Amazon Web Services platform and its AWS Rekognition tool to apply computer vision and artificial intelligence techniques in image analysis. The study was motivated by the curiosity to understand the functioning of mobile device cameras, especially due to the increased use of artificial intelligence and computer vision in these devices in recent years.

Through the use of machine learning algorithms, it is possible to extract impressive results from limited hardware, optimizing resources and performing complex tasks in real time. The real-time identification and recognition of objects, faces, and scenes pave the way for improvements in the quality of photographs and the creation of interactive experiences for users.

Furthermore, the utilization of cloud computing has become a fundamental part of this project, enabling time and local computational resource savings. By combining Python, cloud computing, and artificial intelligence, it was possible to efficiently implement computer vision algorithms, leveraging the scalability and processing power offered by AWS.

Keywords: Machine Learning, Computer Vision, Artificial Intelligence, Cloud Computing

Lista de abreviaturas e siglas

IA Inteligência Artificial

ML Machine Learning

AWS Amazon Web Services

VS Code Visual Studio Code

IDES Integrated Development Environments ou Ambientes Integrados de Desenvolvimento

Sumário

1. Capítulo 1: Introdução.....	14
2. Capítulo 2: Inteligência Artificial	16
2.1. Machine Learning:.....	19
2.2. Visão Computacional	22
3. Python:	25
3.1. VS Code	27
4. Computação em Nuvem:	29
4.1. AWS:.....	34
4.1.1. AWS Rekognition	37
5. Capítulo 3: Projeto de Análise de fotos com AWS Rekognition.....	38
5.1. AWS e Credenciais	38
5.2. Bibliotecas necessárias no código	39
5.3. Criação do cliente da AWS no código	40
5.4. Acessando o rekognition no código.....	41
5.5. Seleção da foto e conversão	42
5.6. Camadas reconhecidas nas fotos, configuração da confiança e verificação de Rostos.....	43
5.7. Armazenamento dos resultados.....	44
5.8. Execução do código.....	45
6. Capítulo 4: Resultados.....	47
6.1. Usando retratos, fotos com pessoas visíveis:.....	47
6.1.1. Foto 1, uma 3x4, até 10 camadas e o mínimo de confiança em 90%	48
6.1.2. Foto 2, usando o piloto de Fórmula 1, Lewis Hamilton como exemplo, até 10 camadas e o mínimo de confiança em 90%	49
6.1.3. Foto 3, apenas um rosto visível, até 10 camadas e o mínimo de confiança em 90%.....	51
6.1.4. Foto 4, 3 rostos visíveis, até 10 camadas e o mínimo de confiança em 90%.....	52

6.2.	Foto 5, 3 rostos não muito visíveis devido à baixa resolução da foto, até 10 camadas e o mínimo de confiança em 90%.....	54
6.3.	Testando o programa com fotos de Paisagem:	56
6.3.1.	Dia de sol na praia, alguns rostos visíveis, até 10 camadas e o mínimo de confiança em 90%:	56
6.3.2.	Foto na neve: sem pessoas, até 10 camadas e o mínimo de confiança em 80%:.....	58
6.3.3.	Miami no nascer do sol: até 20 camadas e o mínimo de confiança em 80%:.....	59
6.3.4.	Dia de carnaval, foto com resolução mais baixa, alguns rostos visíveis, até 10 camadas e o mínimo de confiança em 70%:	61
6.4.	Fotos de Desenho:.....	62
6.4.1.	Desenho 1, 10 camadas e mínimo de confiança em 70%:.....	62
6.4.2.	Desenho 2: 10 camadas e mínimo de confiança em 70%:.....	64
6.4.3.	Desenho 3: 10 camadas e mínimo de confiança em 70%:.....	65
	Figura 53 – Desenho 3	65
6.4.4.	Desenho 4:	66
8.	Referências:.....	69

Lista de Figuras

Figura 1 - Inteligência Artificial.....	16
Figura 2 – Machine Learning	19
Figura 3 – Tipos de Machine Learning.....	20
Figura 4 – Visão Computacional	22
Figura 5 – Python	25
Figura 6 – Logo Visual Studio Code.....	27
Figura 7 – Computação em nuvem	29
Figura 8 – Amazon Web Services	34
Figura 9 – Amazon Rekognition	37
Figura 10 – Tela de Cadastro da AWS.....	39
Figura 11 – Bibliotecas usadas no código.....	39
Figura 12 – Parte do código onde o cliente da AWS é criado localmente.	40
Figura 13 – Parte do código onde o AWS Rekognition é chamado	41
Figura 14 – Parte do código onde é possível escolher uma foto para análise	42
Figura 15 – Parte do código onde é possível configurar o Rekognition ...	43
Figura 16 – Parte do código onde os resultados são exportados para um arquivo .xlsx	44
Figura 17 – Depurador nativo do VS Code	45
Figura 18 – Terminal integrado do VS Code Executando o programa	47
Figura 19 – Foto Com rosto 1	48
Figura 20 – Resultado da Figura 19 no terminal.....	48
Figura 21 – Resultado da Figura 19 no Excel	49
Figura 22 – Foto com rosto 2.....	49
Figura 23 – Resultado da Figura 22 no terminal.....	50
Figura 24 – Resultado da Figura 22 no Excel	50
Figura 25 – Foto com rosto 3.....	51
Figura 26 – Resultado da Figura 25 no terminal.....	51

<i>Figura 27 – Resultado da Figura 25 no Excel</i>	<i>52</i>
<i>Figura 28 – Foto com 3 rostos.....</i>	<i>52</i>
<i>Figura 29 – Resultado da Figura 28 no terminal.....</i>	<i>53</i>
<i>Figura 30 – Resultado da Figura 28 no Excel</i>	<i>53</i>
<i>Figura 31 – Foto de rua feita por celular</i>	<i>54</i>
<i>Figura 32 – Resultado da Figura 31 no terminal.....</i>	<i>54</i>
<i>Figura 33 – Resultado da Figura 31 no Excel</i>	<i>55</i>
<i>Figura 34 – Resultado da Figura 31 no terminal.....</i>	<i>55</i>
<i>Figura 35 – Resultado da Figura 31 no Excel</i>	<i>55</i>
<i>Figura 36 – Foto de paisagem 1</i>	<i>56</i>
<i>Figura 37 – Resultado da Figura 36 no terminal.....</i>	<i>57</i>
<i>Figura 38 – Resultado da Figura 36 no Excel</i>	<i>57</i>
<i>Figura 39 – Foto de paisagem 2</i>	<i>58</i>
<i>Figura 40 – Resultado da Figura 39 no terminal.....</i>	<i>58</i>
<i>Figura 41 – Resultado da Figura 39 no Excel</i>	<i>59</i>
<i>Figura 42 – Foto de paisagem 3</i>	<i>59</i>
<i>Figura 43 – Resultado da Figura 42 no terminal.....</i>	<i>60</i>
<i>Figura 44 – Resultado da Figura 42 no Excel</i>	<i>60</i>
<i>Figura 45 – Paisagem 4.....</i>	<i>61</i>
<i>Figura 46 – Resultado da Figura 45 no terminal.....</i>	<i>61</i>
<i>Figura 47 – Resultado da Figura 45 no Excel</i>	<i>62</i>
<i>Figura 48 – Desenho 1</i>	<i>62</i>
<i>Figura 49 – Resultado da Figura 48 no terminal.....</i>	<i>63</i>
<i>Figura 50 – Resultado da Figura 48 no Excel</i>	<i>63</i>
<i>Figura 51 – Desenho 2</i>	<i>64</i>
<i>Figura 52 – Resultado da Figura 51 no terminal.....</i>	<i>64</i>
<i>Figura 53 – Resultado da Figura 51 no Excel</i>	<i>64</i>
<i>Figura 54 – Desenho 3</i>	<i>65</i>
<i>Figura 55 – Resultado da Figura 54 no terminal.....</i>	<i>65</i>

<i>Figura 56 – Resultado da Figura 54 no Excel</i>	<i>65</i>
<i>Figura 57 – Desenho 4</i>	<i>66</i>
<i>Figura 58 – Resultado da Figura 57 no terminal.....</i>	<i>66</i>
<i>Figura 59 – Resultado da Figura 57 no Excel</i>	<i>67</i>

1. Capítulo 1: Introdução

Esse trabalho teve como primeira e principal inspiração a vontade de entender um pouco mais de como as câmeras dos celulares funcionam, por que de alguns anos para cá tivemos um grande aumento do uso de inteligência artificial e visão computacional nesses pequenos dispositivos, A possibilidade de extrair resultados incríveis a partir de um hardware de tamanho limitado é um exemplo da capacidade de algoritmos de aprendizado de máquina em otimizar recursos e executar tarefas complexas em tempo real. Com o avanço da tecnologia, é possível identificar e reconhecer objetos em imagens, identificar rostos e cenários em tempo real e com isso, é possível aplicar as melhores configurações para obter melhores resultados abrindo caminho para aplicações que vão desde melhorias na qualidade das fotografias até a criação de experiências interativas para os usuários finais. Ao compreender como as câmeras dos celulares funcionam e como a inteligência artificial pode ser aplicada, torna-se possível otimizar o desempenho da captura de imagens e aplicar as melhores configurações para obter resultados cada vez mais impressionantes. Além disso, o uso de visão computacional em tempo real proporciona novas oportunidades para explorar e expandir os limites da fotografia computacional.

A segunda inspiração foi o grande interesse em utilizar os recursos da computação em nuvem. Já que a computação em nuvem é uma das tecnologias mais importantes e transformadoras dos últimos anos. Ao realizar pesquisas sobre a computação em nuvem, foi possível ver que com alguns códigos e bibliotecas já era possível implementar algoritmos de visão computacional alinhados a inteligência artificial, graças a isso, foi possível economizar tempo e poder computacional local, já que se todos esses algoritmos iriam requerer muito poder computacional, o que provavelmente não seria possível com um simples computador e exigiria muito tempo também para treinar esses mesmos algoritmos fossem criados e implementados de forma local. Ao estudar mais sobre essa tecnologia, foi possível perceber como a combinação de computação em nuvem e inteligência artificial pode ser extremamente útil para a solução de

problemas complexos. Com a utilização de algoritmos de visão computacional e inteligência artificial em nuvem, foi possível economizar tempo e recursos computacionais locais, uma vez que esses algoritmos demandam grande poder de processamento e treinamento de dados. Assim, a combinação de Python, Computação em nuvem e inteligência artificial foi essencial para o sucesso do projeto. Para complementar, é importante destacar que a visão computacional e a inteligência artificial têm se tornado cada vez mais populares em diversas áreas, desde aplicativos de redes sociais até em indústrias e sistemas de segurança. O uso de técnicas de machine learning e deep learning em imagens tem permitido que computadores aprendam a identificar e classificar objetos com alta precisão, o que pode ser usado para análise de dados, reconhecimento de padrões, diagnósticos médicos, automação industrial e muito mais. Além disso, a computação em nuvem tem permitido a democratização do acesso a recursos de alta performance, permitindo que desenvolvedores e pesquisadores possam implementar e testar algoritmos em larga escala e sem a necessidade de grandes investimentos em infraestrutura.

Dentre as 3 grandes plataformas de computação em nuvem (Google, Amazon e Microsoft) a plataforma da Amazon foi a escolhida por já ter uma ferramenta específica para análise de imagens e também devido à questões de menor custos já que alguns recursos podem ser usados de forma gratuita para testes, apenas com uma conta padrão e também por oferecer uma maior facilidade de usar os recursos. Além disso, a escolha pela plataforma da Amazon, também conhecida como AWS (Amazon Web Services), se deu por conta da ampla documentação e comunidade de usuários, que possibilitam um suporte mais amplo e acesso a soluções já implementadas por outras empresas e desenvolvedores. Neste trabalho, a plataforma AWS Rekognition foi utilizada como uma ferramenta poderosa para análise de imagens, que com suas capacidades avançadas de reconhecimento e análise oferece diversos recursos, como reconhecimento de objetos, faces, texto e outras características. Essa ferramenta é baseada em tecnologia de aprendizado de máquina, o que possibilita um alto nível de precisão e eficiência na análise das imagens.

Ao escolher a plataforma da Amazon e utilizar a ferramenta AWS Rekognition, o desenvolvedor tem à sua disposição uma vasta gama de recursos

e ferramentas para trabalhar com inteligência artificial e análise de imagens de forma eficiente e escalável, podendo desenvolver soluções para diversas áreas, como segurança, saúde, varejo, entre outras.

No decorrer deste trabalho, foi realizado um experimento com testes práticos para explorar as capacidades do AWS Rekognition simulando as análises que os celulares fazem reconhecendo cenários, rostos e objetos.

2. Capítulo 2: Inteligência Artificial



Figura 1 - Inteligência Artificial (Imagem gerada por inteligência Artificial: BlueWillow)

Inteligência Artificial (IA) é um campo de estudo que envolve a criação de sistemas e tecnologias capazes de executar tarefas que, até então, só poderiam ser realizadas por seres humanos. Essas tarefas incluem o aprendizado, o raciocínio, a percepção e a resolução de problemas. A I.A também pode ser definida como a pesquisa baseada na melhora de processos humanos em áreas como educação, agricultura, saúde.

“A Inteligência artificial surgida na década de 1950 tem sua origem praticamente confundida com a própria origem do computador. Mais precisamente, no verão de 1956, ocorreu a Dartmouth College Conference, que é considerada o marco inicial da IA. Os pesquisadores reconhecidos como pais

da área, como John MacCarthy, Marvin Minsky, Alan Newell e Herbert Simon, entre outros, participaram desse evento e tiveram trajetórias científicas que estabeleceram marcos nesse fascinante domínio da Computação.” (Jaime Simão Sichman, 2021)

Um dos humanos é um interrogador que fica separado do outro humano e do sistema de IA. Este interrogador entra em uma conversa em linguagem natural (via teclado) com o outro humano e também com a máquina, e caso ele não consiga distinguir se está conversando com a máquina ou com o ser humano é um indicativo de que o sistema é inteligente e passou no Teste de Turing. (GRANATYR, 2016)

Em 1959, surge pela primeira vez o termo machine learning, descrevendo um sistema que dá aos computadores a habilidade de aprender alguma função sem serem programados diretamente pra isso. Basicamente, significa alimentar um algoritmo com dados, para que a máquina aprenda a executar uma tarefa automaticamente (INSTITUTO DE ENGENHARIA, 2016).

Passando por análises e descobertas, em 1964 a I.A. deu outro passo importante em sua história, quando criaram o primeiro chatbot do mundo. Com nome de ELIZA, sua função era realizar conversas de forma automática, comandada por dados e algoritmos que eram baseados em palavras-chave imitando uma psicanalista (INSTITUTO DE ENGENHARIA, 2016).

A IA é uma tecnologia que tem ganhado cada vez mais espaço em nossa sociedade. Ela está presente em nossas casas, em nossos celulares, em nossos carros, em nossos trabalhos, entre outros lugares. Isso se deve ao fato de que a IA é capaz de processar e analisar grandes quantidades de dados de forma rápida e precisa, o que permite a criação de soluções e insights valiosos em diversas áreas

A IA é composta por diversas técnicas e subáreas, como aprendizado de máquina, processamento de linguagem natural, visão computacional, robótica, entre outras. Cada uma dessas subáreas busca resolver problemas específicos e, juntas, formam a base da IA.

O aprendizado de máquina é uma das técnicas mais utilizadas na IA. Ele consiste em ensinar um sistema a identificar padrões em dados e tomar decisões com base nesses padrões. O sistema "aprende" a partir de exemplos, o que o torna capaz de reconhecer esses padrões em novos dados e tomar decisões de forma autônoma. Os algoritmos de aprendizado de máquina são projetados para "aprender" a partir de dados de entrada e, assim, ser capazes de prever ou tomar decisões com base em novos dados. Esses algoritmos são divididos em duas categorias principais: supervisionados e não supervisionados. O processamento de linguagem natural é outra subárea da IA que está em alta. Ela permite que os sistemas entendam e processem a linguagem humana, o que possibilita a criação de chatbots, assistentes virtuais e sistemas de tradução automática. A visão computacional é outra subárea importante da IA, que permite a análise e interpretação de imagens e vídeos. Ela é utilizada em diversas aplicações, como reconhecimento facial, análise de imagens médicas e veiculares, e em sistemas de vigilância e segurança.

A robótica é uma área que tem se beneficiado bastante da IA. Ela permite a criação de robôs autônomos capazes de realizar tarefas complexas, como navegação em ambientes desconhecidos e interação com humanos. A IA tem o potencial de transformar a forma como vivemos e trabalhamos, e de gerar grandes avanços em áreas como saúde, transporte, comunicação e educação. No entanto, é importante destacar que a IA também traz desafios e preocupações, como o impacto na economia e no mercado de trabalho, a privacidade e a segurança dos dados, e a ética no uso dessas tecnologias. Por isso, é fundamental que a IA seja desenvolvida de forma responsável e ética, levando em conta os impactos sociais e culturais que ela pode gerar.

2.1. Machine Learning:

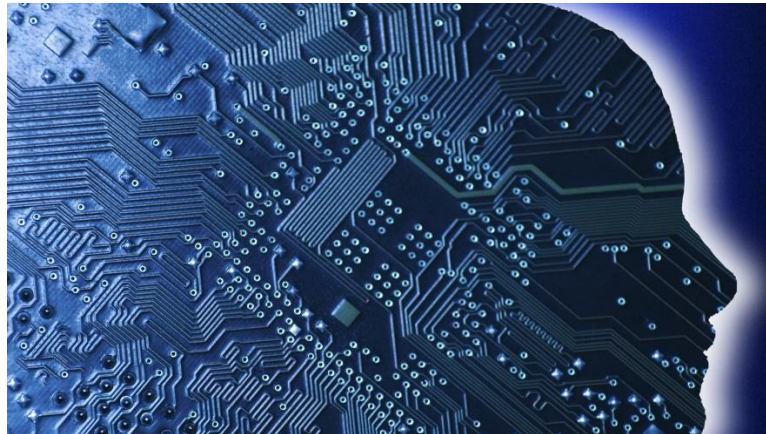


Figura 2 – Machine Learning (Fonte: Google Imagens)

A Aprendizagem de Máquina (ou Machine Learning) é uma das tendências mais modernas da tecnologia atualmente. É uma área da Inteligência Artificial relacionada à busca de um conjunto de regras e procedimentos para permitir que as máquinas possam agir e tomar decisões baseadas em dados, ao invés de serem programadas para realizar uma determinada tarefa. (Arthur Samuel, 1959). A Aprendizagem de Máquina vem se tornando cada vez mais importante nos últimos anos, graças aos avanços tecnológicos e ao aumento da disponibilidade de dados. O objetivo é ensinar as máquinas a reconhecer padrões nos dados e, com isso, serem capazes de fazer previsões e tomar decisões autônomas de forma mais precisa e eficiente.

Warren McCulloch e Walter Pitts apresentaram um artigo que falava de redes neurais em 1943. Já em 1950 Claude Shannon programou uma máquina para poder jogar xadrez com cálculos de posição simples, porém eficientes assim exibindo o que a IA já poderia fazer.

A aprendizagem de máquina é um subconjunto da inteligência artificial, o segmento da ciência da computação que se concentra na criação de computadores que pensam da maneira que os humanos pensam. Em outras palavras, todos os sistemas de aprendizado de máquinas são sistemas de IA, mas nem todos os sistemas de IA possuem capacidades de aprendizado de máquina. O aprendizado de máquina pode ser dividido em diferentes categorias:

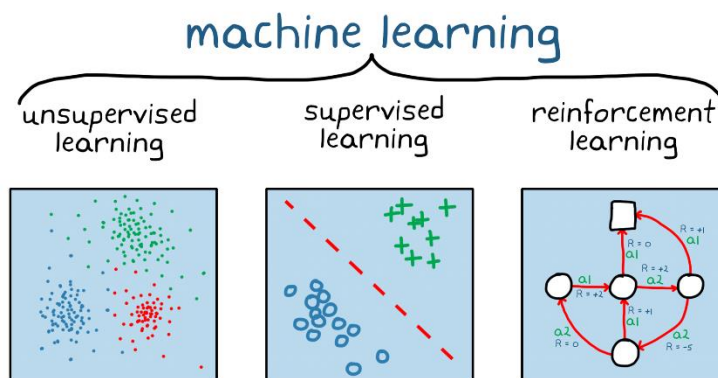


Figura 3 – Tipos de Machine Learning (Fonte: Google Imagens)

- **Aprendizagem Supervisionada** – o aprendizado supervisionado requer um programador ou “professor” que ofereça exemplos de quais entradas se alinham com os resultados. Por exemplo, se você quisesse usar a aprendizagem supervisionada para ensinar um computador a reconhecer fotos de gatos, você forneceria um conjunto de imagens, algumas que foram rotuladas como “gatos” e algumas que foram rotuladas como “não gatos”. Os algoritmos de aprendizagem de máquina ajudariam o sistema a aprender a generalizar os conceitos para que ele pudesse identificar gatos em imagens que não havia encontrado antes, ou seja, em novos conjuntos de dados.
- **Aprendizagem Não-Supervisionada** – A aprendizagem não supervisionada exige que o sistema desenvolva suas próprias conclusões a partir de um determinado conjunto de dados. Por exemplo, se você tivesse um grande conjunto de dados de vendas on-line, você poderia usar a aprendizagem sem supervisão para encontrar clusters ou associações entre esses dados que poderiam ajudá-lo a melhorar seu marketing. Você pode descobrir, por exemplo, que as mulheres nascidas no início da década de 1980 com renda superior a 5 mil reais por mês, têm afinidade por uma marca particular de barra de chocolate ou que as pessoas que compram uma determinada marca de refrigerante também compram uma certa marca de chips.
- **Aprendizagem Semi-Supervisionada** – A aprendizagem semi-supervisionada, como você provavelmente adivinhou, é uma combinação de aprendizado supervisionado e não supervisionado.

Voltando ao exemplo do gato, imagine que você tenha muitas imagens, algumas das quais foram rotuladas como “gato” e “não gato” e algumas outras imagens sem rótulos. Um sistema de aprendizagem semisupervisionado usaria as imagens rotuladas para fazer inferências sobre qual das imagens não marcadas inclui gatos. As melhores suposições seriam então devolvidas ao sistema para ajudá-lo a melhorar suas capacidades e o ciclo continuaria.

- **Aprendizagem Por Reforço** – o aprendizado por reforço envolve um sistema que recebe feedback análogo a punições e recompensas. Um exemplo clássico de aprendizagem de reforço (como se aplica à aprendizagem de máquina) é um agente aprendendo a jogar um game. O objetivo é vencer o game e o agente vai sendo recompensado ou punido de acordo com seus erros e acertos, até atingir seu objetivo.

A aplicação da Aprendizagem de Máquina é vasta, abrangendo áreas como finanças, saúde, marketing, automação industrial, entre outras. O uso de algoritmos de Aprendizagem de Máquina permite automatizar processos complexos, aumentar a eficiência e reduzir custos. Além disso, o aprendizado de máquina tem o potencial de descobrir novos padrões e insights que seriam difíceis ou impossíveis de se obter manualmente, contribuindo para avanços científicos e tecnológicos significativos.

Além disso, a aprendizagem de máquina tem sido uma das principais impulsionadoras do desenvolvimento de outras tecnologias relacionadas à inteligência artificial, como a visão computacional e o processamento de linguagem natural. Isso tem permitido o desenvolvimento de sistemas mais avançados, capazes de interpretar imagens e textos com maior precisão.

2.2. Visão Computacional

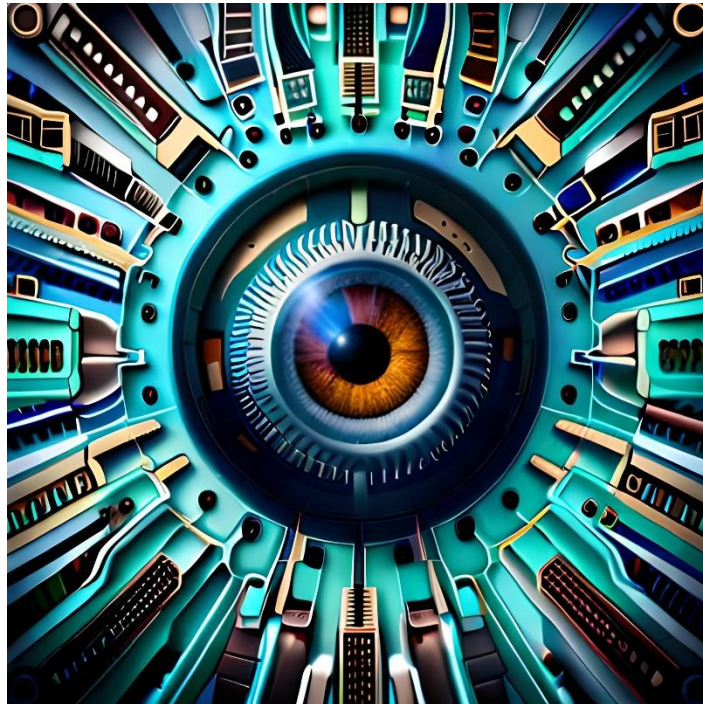


Figura 4 – Visão Computacional (Imagem gerada por inteligência Artificial: BlueWillow)

“Visão computacional é a ciência responsável pela visão de uma máquina, pela forma como um computador enxerga o meio à sua volta, extraindo informações significativas a partir de imagens capturadas por câmeras de vídeo, sensores, scanners, entre outros dispositivos. Estas informações permitem reconhecer, manipular e pensar sobre os objetos que compõem uma imagem” (Azevedo Conci Leta, 2008)

A visão computacional é uma área da inteligência artificial e processamento de imagem que tem como objetivo capacitar as máquinas a interpretar e compreender visualmente o mundo ao seu redor. Uma das primeiras menções sobre visão computacional ocorreu em 1955 onde Selfridge destacou “...eyes and ears for the computer” e por volta da década de 70 foi iniciado os primeiros trabalhos de Visão Computacional aliada a Inteligência Artificial. Durante os anos 1980 e 1990, ocorreram avanços significativos na área, impulsionados pelo desenvolvimento de técnicas de processamento de imagem e aprendizado de máquina. O surgimento de algoritmos baseados em redes neurais artificiais, como as redes neurais convolucionais, permitiu melhorias

significativas na capacidade dos sistemas de visão computacional em reconhecer e classificar objetos em imagens. Nos últimos anos, a visão computacional tem sido impulsionada pelo aumento da capacidade de processamento, avanços na área de aprendizado profundo e disponibilidade de conjuntos de dados cada vez maiores. Isso permitiu o desenvolvimento de sistemas de visão computacional mais precisos e eficientes, capazes de realizar tarefas cada vez mais complexas.

A visão computacional busca replicar a forma como os seres humanos enxergam e interpretam o ambiente ao seu redor por meio de algoritmos, técnicas avançadas de processamento de imagem utilizando combinação de aprendizado de máquina e análise estatística. Os sistemas de visão computacional podem realizar uma série de tarefas, como detecção e reconhecimento de objetos, identificação de padrões, segmentação de imagens, medição de distâncias e até mesmo compreensão de gestos e expressões faciais. Os algoritmos são treinados com conjuntos de dados para reconhecer padrões específicos e realizar tarefas específicas. Esses dados podem incluir imagens rotuladas, onde cada objeto de interesse é identificado e categorizado.

No contexto da computação em nuvem, plataformas como a AWS oferecem serviços específicos, como o AWS Rekognition, que permitem a aplicação da visão computacional de forma escalável e acessível. Isso possibilita que desenvolvedores e empresas utilizem essas poderosas capacidades sem precisar investir em infraestrutura e recursos computacionais.

Com o avanço da tecnologia e o desenvolvimento de novos algoritmos de visão computacional, cada vez mais recursos estão sendo incorporados às câmeras dos celulares, permitindo que os usuários capturem e editem imagens de forma mais criativa e profissional. Tais como:

- **HDR (High Dynamic Range):** A técnica de HDR combina várias exposições de uma mesma cena para criar uma imagem final com maior faixa dinâmica. Isso significa que detalhes em áreas de sombras e destaques são preservados, resultando em imagens mais equilibradas e realistas.

- **Modo Retrato:** O modo retrato utiliza algoritmos de segmentação de imagem para desfocar o fundo e destacar o objeto principal, criando um efeito semelhante ao obtido com câmeras profissionais. Isso permite criar retratos com fundo desfocado, dando ênfase ao objeto ou pessoa fotografada.
- **Deteção de faces:** Os algoritmos de detecção de faces permitem que os celulares identifiquem rostos nas imagens e apliquem automaticamente recursos como foco, equilíbrio de branco e ajuste de exposição para obter melhores resultados em retratos. Além disso, esses algoritmos também podem ser utilizados para reconhecimento facial e autenticação biométrica.
- **Modo de captura de movimento:** Alguns celulares oferecem modos especiais de captura de movimento, nos quais múltiplas imagens são capturadas em sequência e combinadas para criar uma imagem final com a aparência de movimento. Isso é especialmente útil em fotografias esportivas ou em situações em que há movimento intenso.
- **Melhoria de pouca luz/ Modo noturno:** Os celulares geralmente enfrentam desafios ao capturar imagens em condições de pouca luz. No entanto, com técnicas de fotografia computacional, é possível reduzir o ruído e melhorar a qualidade das imagens capturadas nessas condições. Algoritmos de fusão de imagens e ajuste de exposição são utilizados para produzir resultados mais nítidos e com menos ruído.
- **Reconhecimento de cenas:** Os celulares podem usar algoritmos de aprendizado de máquina para identificar automaticamente o tipo de cena que está sendo fotografada, como paisagens, praias, pôr do sol, comida, entre outros. Com base nessa análise, os ajustes de câmera podem ser otimizados para obter melhores resultados em cada tipo de cena.

3. Python:



Figura 5 – Python (Fonte: <https://www.python.org/>)

Foi Criada por Guido van Rossum em 1990 no Instituto Nacional de Pesquisa para Matemática e Ciência da Computação da Holanda e tinha como foco originalmente físicos e engenheiros e foi lançada em 1991 e desde então tornou-se uma das linguagens de programação mais populares do mundo. O Python é amplamente utilizado em uma variedade de campos, incluindo aprendizado de máquina, análise e ciência de dados, desenvolvimento web, inteligência artificial, banco de dados, animações 3D, aplicativos móveis, jogos, automação de tarefas entre outros. É conhecido por ser uma linguagem fácil de aprender devido à sua sintaxe clara e concisa que se assemelha muito ao inglês. A comunidade Python é grande e ativa oferecendo uma ampla gama de bibliotecas, módulos e frameworks, isso explica o porquê podemos falar que Python é “batteries included” ou seja, baterias inclusas, fazendo alusão a um produto completo que pode ser usado prontamente.

Python é uma linguagem de programação versátil e amplamente utilizada, conhecida por sua simplicidade e legibilidade de código. Uma das vantagens mais notáveis do Python é o fato de ser um software livre e gratuito, o que significa que qualquer pessoa pode utilizá-lo, modificá-lo e distribuí-lo sem restrições. Essa característica é resultado do trabalho dedicado da Python Software Foundation, uma organização sem fins lucrativos, e da contribuição contínua de uma comunidade global de desenvolvedores. No ecossistema do Python, existem várias opções de ambientes integrados de desenvolvimento (IDEs) disponíveis para auxiliar os programadores no processo de escrita, edição

e depuração de código. Algumas das IDEs populares incluem o Visual Studio Code, PyCharm e Jupyter Notebook. Essas ferramentas oferecem recursos avançados, como realce de sintaxe, depuração interativa, sugestões de código e gerenciamento de projetos, tornando o desenvolvimento em Python mais eficiente e produtivo. Além das IDEs, também é possível utilizar editores de texto com suporte para Python. Existem várias opções disponíveis, como o Sublime Text, Atom, Notepad++, entre outros. Esses editores fornecem recursos básicos de edição de texto, como destaque de sintaxe, recuo automático e pesquisa e substituição avançadas, permitindo que os desenvolvedores escrevam e editem código Python de forma conveniente. Uma das grandes vantagens do Python é a sua portabilidade. Ele pode ser executado em praticamente qualquer arquitetura de computador ou sistema operacional, incluindo Linux, Windows, macOS e outros. Isso significa que os programas escritos em Python podem ser facilmente compartilhados e executados em diferentes ambientes sem a necessidade de realizar grandes alterações no código. Essa flexibilidade contribui para a ampla adoção do Python em diversos cenários, desde desenvolvimento web e científico até automação de tarefas e inteligência artificial.

Em resumo, o Python oferece um conjunto diversificado de ferramentas de desenvolvimento, desde IDEs poderosas até editores de texto com suporte para Python. Sua natureza de software livre e gratuito, juntamente com sua portabilidade, fazem dele uma escolha popular entre os desenvolvedores. Com uma comunidade ativa e em constante crescimento, o Python continua evoluindo e se tornando uma das linguagens de programação mais populares e influentes no mundo da tecnologia.

Tem algumas características interessantes:

- **Sua interpretação:** por ser uma linguagem interpretada o Python não precisa ser compilado antes de ser executado o que acaba tornando o desenvolvimento mais rápido e eficiente.
- **Tipagem dinâmica:** Essa linguagem não requer a declaração explícita de variáveis. O tipo da variável é determinado dinamicamente em tempo de execução.

- **Orientação a objetos:** O Python suporta programação orientada a objetos, o que permite que códigos possam ser modularizados, utilizando classes e objetos.

3.1. VS Code

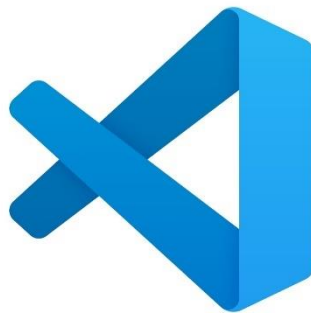


Figura 6 – Logo Visual Studio Code (Fonte: <https://code.visualstudio.com/>)

O Visual Studio Code ou VS Code como é comumente conhecido é um editor de código fonte, multiplataforma, open source oferecido de forma gratuita desenvolvido pela Microsoft lançado em 2015. O VSCode oferece uma plataforma flexível e poderosa para escrever, editar e depurar código e é capaz de atender uma quantidade enorme de projetos (ASP .NET, Node.js) e oferece suporte a mais de 30 linguagens de programação como Python, JavaScript, C#, C++, PHP, Java, HTML, R, CSS, SQL, Markdown, TypeScript, LESS, SASS, JSON, XML assim como muitos outros formatos de arquivos comuns.

Uma das principais características do VS Code é sua interface intuitiva e amigável. Com uma aparência moderna e limpa, o editor proporciona uma experiência agradável ao usuário, tornando a programação mais eficiente e produtiva. Além disso, o VS Code é altamente personalizável, permitindo que os desenvolvedores escolham entre uma variedade de temas, extensões e configurações para adaptar o ambiente às suas necessidades específicas. Além disso, o editor também oferece suporte a recursos avançados, como realce de sintaxe, autocompletar código, formatação automática, sugestões de código e

integração com sistemas de controle de versão, facilitando o desenvolvimento de software em diferentes contextos.

Outro destaque do VS Code é a sua extensa coleção de extensões. Os desenvolvedores podem aproveitar uma grande variedade de extensões disponíveis no marketplace do VS Code para estender a funcionalidade do editor. Essas extensões podem adicionar recursos específicos de uma linguagem de programação, ferramentas de depuração avançadas, integração com serviços em nuvem, suporte a frameworks e bibliotecas populares, entre outras funcionalidades personalizadas.

O VS Code possui recursos poderosos de depuração, permitindo que os desenvolvedores identifiquem e corrijam erros em seus programas de forma eficiente. Com suporte para depuração em várias linguagens, breakpoints, inspeção de variáveis, controle de execução passo a passo e outras funcionalidades, o VS Code torna o processo de depuração mais acessível e facilita a identificação e solução de problemas em tempo real. O VS Code também oferece recursos avançados de produtividade, como controle integrado de terminal, edição em várias colunas, busca e substituição avançada, controle de git integrado, entre outros. Esses recursos permitem que os desenvolvedores realizem tarefas comuns de forma rápida e eficiente, aprimorando ainda mais a experiência de desenvolvimento.

A seguir, um resumo dos principais recursos desta ferramenta (MACORATTI,2016):

- **Edição focalizada em código:** Um editor com gestos centralizados no teclado, significa que não é preciso alcançar o mouse. Possibilita abrir arquivos grandes ou pequenos instantaneamente no código sem distrações. Também possui recursos de edição avançados, tais como cursores múltiplos, salvamento automático e muito mais.
- **Navegação de código:** Permite que o programador permaneça no contexto enquanto se move através de grandes arquivos de código e através da sua base de código. A barra de rolagem realça os problemas no arquivo atual, a localização integrada suporta expressões regulares.

- **Compreensão de código:** O *IntelliSense* descreve as APIs enquanto o programador digita, com preenchimento automático para aumentar a velocidade e a precisão. As ferramentas de informações rápidas permitem que inspecionar definições de API, e os “rabiscos” informam sobre problemas enquanto digita.
- **Depuração:** É possível definir pontos de interrupção no código, interromper exceções, inspecionar variáveis, percorrer o código, navegar pela pilha de chamadas.
- **Controle de versão do Git:** Faz conexão ao universo Git inteiro extraíndo e enviando código para qualquer editor remoto – GitHub, Azure Web Apps, Visual Studio Team Services e muito mais. Possibilita a visualização de arquivos alterados, comparação e realização de confirmações direto do editor.

4. Computação em Nuvem:

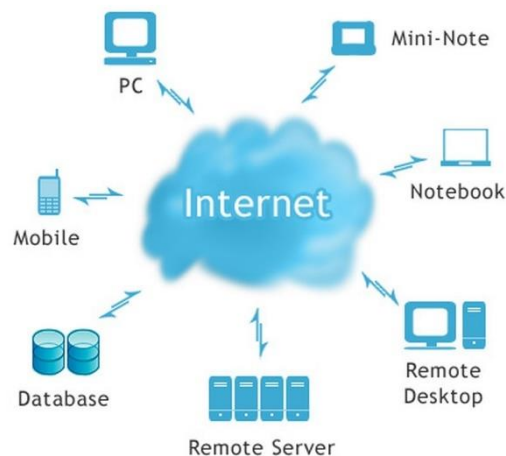


Figura 7 – Computação em nuvem (Fonte: Google Images)

A Computação em nuvem ou ‘Cloud Computing’ é um modelo de computação que permite o acesso a um grande conjunto de recursos, aplicações e serviço de computação que são compartilhados, configurados e podem ser solicitados e fornecidos rapidamente com o menor esforço possível por parte do usuário desde que esteja conectado a rede da nuvem, demandando mínima

interação com o provedor de serviços. Os recursos disponíveis são disponibilizados de forma sob demanda para o usuário. Empresas como Amazon, Google, Microsoft, Oracle e IBM foram as pioneiras nessa grande ofensiva de computação em nuvem que atualmente é dividida em 10 modelos de serviços:

- **IaaS:** “Infrastructure as a Service” ou Infraestrutura como Serviço: É um modelo de serviço em nuvem que oferece recursos de infraestrutura sob demanda altamente escalonáveis como serviços na Internet. As nuvens de IaaS geralmente oferecem recursos adicionais, como uma biblioteca de imagem de disco de máquina virtual, armazenamento bruto de bloco, armazenamento de arquivos ou objetos, firewalls, balanceadores de carga, endereços IP, VLANs (redes locais virtuais) e pacotes de software. Os provedores de nuvem IaaS fornecem esses recursos sob demanda a partir de seus grandes pools de equipamentos instalados nos datacenters. Para conectividade de área ampla, os clientes podem usar a Internet ou as nuvens da operadora (redes privadas virtuais dedicadas). Para implantar seus aplicativos, os utilizadores da nuvem instalam imagens do sistema operacional e seu software de aplicativo na infraestrutura de nuvem. Nesse modelo, o utilizador da nuvem corrige e mantém os sistemas operacionais e o software do aplicativo. Tendo como benefícios o alto desempenho, confiabilidade, preços competitivos e a possibilidade de ter backup e recuperação.
- **PaaS:** “Platform as a Service” ou Plataforma como Serviço: É uma categoria de serviço de computação em nuvem que oferece uma plataforma completa para o desenvolvimento, hospedagem, execução e gerenciamento de aplicativos sem que o usuário precise se preocupar com a infraestrutura subjacente. Essa categoria de serviço oferece uma camada de abstração que simplifica o processo de desenvolvimento e implantação de aplicativos, permitindo que os desenvolvedores se concentrem no desenvolvimento do aplicativo em vez de gerenciar a infraestrutura. Os serviços PaaS geralmente incluem recursos como provisionamento de recursos, gerenciamento de bancos de dados, monitoramento de aplicativos, escalabilidade automática e suporte a

vários idiomas de programação. Esses recursos podem ser usados para criar e implantar aplicativos em uma variedade de plataformas, como a web, dispositivos móveis e Internet das Coisas. Com o PaaS, os usuários têm acesso a um conjunto de ferramentas de desenvolvimento, bibliotecas e frameworks que podem ser usados para criar aplicativos sem precisar se preocupar com as preocupações de infraestrutura.

- **SaaS:** “Software as a Service” ou Software como Serviço é o tipo de serviço online mais conhecido e utilizado como por exemplo por serviços de e-mail marketing ou softwares do G Suite (Google suite) que oferece diversas ferramentas de comunicação, armazenamento de arquivos e criação e compartilhamento de documentos. Nesse modelo, o provedor de serviços é responsável por gerenciar a infraestrutura, manter o software atualizado e oferecer suporte técnico ao cliente. O usuário final pode acessar o software por meio de uma interface de usuário amigável, geralmente em um navegador da web, sem precisar instalar o software em seu próprio computador ou dispositivo. O SaaS tornou-se uma opção popular para empresas que procuram reduzir os custos de infraestrutura de TI e aumentar a eficiência operacional. Isso ocorre porque o SaaS permite que as empresas acessem tecnologias de ponta sem precisar investir em infraestrutura física, o que pode ser caro e demorado. Além disso, o SaaS geralmente opera em um modelo de assinatura, o que significa que as empresas podem pagar apenas pelo que usam e podem escalar rapidamente quando precisam de mais recursos. Como os provedores de serviços de SaaS são responsáveis pela manutenção do software, eles geralmente fornecem atualizações regulares de recursos e correções de bugs, garantindo que o software esteja sempre atualizado e pronto para uso.
- **CaaS:** “Containers as a Service” ou Contêineres como serviço: É um modelo de serviço em nuvem que oferece ao usuário o serviço de virtualização baseado em contêineres. A principal diferença entre CaaS e outras ofertas em nuvem é que os contêineres fornecem uma unidade de execução para aplicativos, permitindo que eles sejam executados em vários ambientes sem alterações significativas. Com CaaS, os usuários

têm acesso a uma plataforma que lhes permite implantar, executar e gerenciar contêineres sem precisar lidar com a complexidade do gerenciamento de infraestrutura subjacente. Isso permite que as empresas se concentrem em seus aplicativos e serviços, em vez de gerenciar infraestrutura. Além disso, o CaaS também ajuda as empresas a economizar em custos de infraestrutura, já que o modelo de pagamento é baseado no uso, permitindo que os usuários paguem apenas pelo que usam. Isso é particularmente útil para empresas que têm picos de tráfego sazonais ou que precisam escalar rapidamente para atender à demanda do mercado.

- **DaaS:** “Desktop as a Service” ou Área de trabalho como Serviço: É uma solução de computação em nuvem na qual a infraestrutura de desktop virtual é terceirizada para um provedor terceirizado. Em outras palavras, é a oferta de um ambiente de trabalho completo, com sistema operacional, aplicativos e dados, hospedado em um servidor remoto e acessível a partir de qualquer dispositivo com acesso à internet. O DaaS é uma das várias formas de fornecer desktops virtuais, que podem ser uma alternativa mais econômica e segura para as empresas do que a aquisição e manutenção de hardware e software. Além disso, o DaaS permite maior flexibilidade e mobilidade para os usuários, já que é possível acessar o ambiente de trabalho a partir de qualquer dispositivo e localização, pois o DaaS usa execução distribuída ou execução remota, dependendo do tipo de implementação. As soluções de DaaS são fornecidas por provedores de serviços em nuvem, que gerenciam o ambiente de desktop virtual, incluindo atualizações, manutenção e suporte técnico. Isso permite que as empresas reduzam custos e se concentrem em suas atividades principais, enquanto o provedor de serviços se responsabiliza pela infraestrutura e gerenciamento do ambiente de desktop virtual.
- **XaaS:** “Everything as a Service” ou Tudo como serviço: Quando se utiliza tudo, infraestrutura, plataformas, software, suporte, e outros como um Serviço. Isso significa que, em vez de adquirir e manter recursos físicos, como servidores, softwares ou equipamentos de rede, o cliente pode

contratar esses serviços pela internet, pagando apenas pelo que usa e sem precisar se preocupar com manutenção ou atualização de infraestrutura. Dentre os tipos de serviços oferecidos em um modelo XaaS, podemos citar: Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS), Function as a Service (FaaS), Desktop as a Service (DaaS), Database as a Service (DBaaS), entre outros. O modelo XaaS oferece diversas vantagens para empresas de todos os tamanhos e segmentos. Além de permitir a redução de custos, ao eliminar a necessidade de aquisição e manutenção de recursos físicos, também oferece maior flexibilidade e escalabilidade, permitindo que a empresa contrate apenas o que precisa em determinado momento e aumente ou diminua o uso de recursos de acordo com sua demanda.

- **FaaS:** “Function as a Service” ou Função como Serviço: É uma chamada de procedimento remoto hospedada em um serviço de computação em nuvem que aproveita a computação sem servidor para poder permitir a criação, implementação e execução de funções individuais na nuvem que são executadas em resposta a eventos programados pelo usuário onde cada função é uma peça independente de código, e é considerado um modelo “serveless” pois os desenvolvedores não precisam provisionar e gerenciar servidores para implantar seus aplicativos. Os provedores de serviços em nuvem, como a Amazon Web Services, o Google Cloud Platform e o Microsoft Azure, oferecem plataformas FaaS que fornecem recursos para desenvolvedores criarem, testarem e implantarem suas funções sem precisar gerenciar servidores ou infraestrutura. Essas plataformas geralmente incluem ferramentas de monitoramento, gerenciamento de logs e escalabilidade automática para lidar com picos de tráfego.
- **DbaaS:** “Database as a Service” ou Banco de Dados como Serviço: Direcionada ao fornecimento de serviços para armazenamento e acesso de volumes de dados. A vantagem aqui é que o detentor da aplicação conta com maior flexibilidade para expandir o banco de dados modelo de serviço em nuvem que permite o acesso a dados em tempo real sem a necessidade de armazená-los localmente. Esse serviço é oferecido por

provedores de nuvem que gerenciam e fornecem acesso a grandes volumes de dados em um formato facilmente acessível e utilizável sem a necessidade de gerenciá-lo fisicamente. Com o DBaaS, o provedor de nuvem é responsável por todas as tarefas relacionadas à configuração, manutenção e otimização do banco de dados. Isso inclui o monitoramento da performance, a aplicação de patches de segurança, a configuração de backups e muito mais. Dessa forma, os usuários podem se concentrar em usar os dados e aplicativos que dependem do banco de dados, em vez de se preocupar com a infraestrutura por trás disso.

4.1. AWS:



Figura 8 – Amazon Web Services (Fonte: <https://aws.amazon.com/pt/>)

AWS ou Amazon Web Services é uma plataforma líder em serviços em nuvem oferecida pela Amazon. Lançada em 2006, a AWS rapidamente se tornou uma das plataformas mais populares e amplamente adotadas no mundo da computação em nuvem. Com uma abordagem abrangente e escalável, a AWS oferece uma vasta gama de serviços em nuvem para atender às necessidades

de empresas de todos os tamanhos e setores. Esses serviços abrangem diversas áreas, incluindo computação, armazenamento de dados, processamento de dados, banco de dados, hospedagem de aplicativos, segurança, análise de dados, Internet das Coisas (IoT), machine learning, inteligência artificial e muito mais.

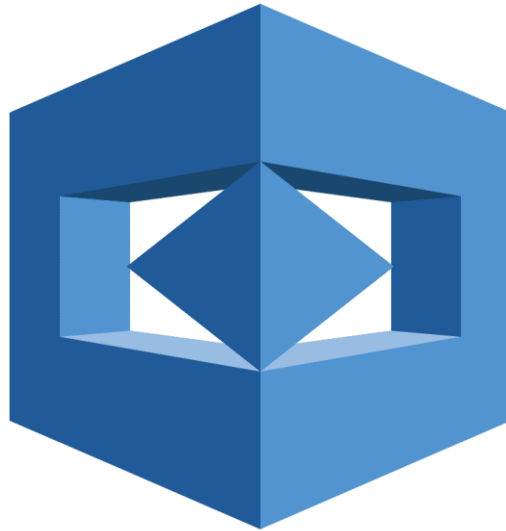
É conhecida por ser a pioneira em implementar o modelo de nuvem 'pague somente pelo que utiliza' ou pay-as-you-go. Esse serviço tem uma utilidade muito importante para seus clientes pois promete ajudar a reduzir custos empresariais e inovarem nos seus serviços com maior agilidade, sem a necessidade de investir em infraestrutura física ou enfrentar custos de manutenção. A plataforma também é altamente segura, com recursos de segurança e conformidade que garantem a proteção dos dados dos usuários. Outro ponto forte da AWS é a sua capacidade de inovação constante, com o lançamento regular de novos serviços e atualizações de recursos existentes. Além disso, a plataforma é altamente integrável com outras tecnologias e ferramentas, permitindo aos usuários criar soluções personalizadas e integrá-las facilmente à infraestrutura da AWS. A plataforma combina modelos de serviços Cloud como:

Infraestrutura como serviço (IaaS), Plataforma como serviço (PaaS), Pacote de software como serviço (SaaS).

A AWS possui uma arquitetura única baseada em zonas de disponibilidade distribuídas geograficamente. Essas zonas de disponibilidade são compostas por data centers físicos estrategicamente localizados em diferentes áreas do mundo. Cada zona de disponibilidade é um ambiente isolado e autônomo, com recursos de computação, armazenamento e rede independentes. Essa distribuição geográfica das zonas de disponibilidade é um dos principais diferenciais da AWS. Ela permite que os usuários tenham alta disponibilidade e tolerância a falhas em seus serviços e aplicações. Em caso de falha em um dos data centers em uma zona de disponibilidade, a AWS é capaz de redirecionar automaticamente o tráfego para outro data center em uma zona de disponibilidade diferente, garantindo a continuidade dos serviços sem perda de desempenho ou interrupção para os usuários. Além disso, a distribuição

geográfica dos data centers da AWS oferece benefícios adicionais. Os usuários podem implantar seus serviços em diferentes regiões do mundo, permitindo que eles sejam acessados com baixa latência por usuários localizados em diferentes áreas geográficas. Essa capacidade é especialmente importante para empresas com atuação global, que podem disponibilizar seus aplicativos e serviços em locais estratégicos próximos aos seus clientes, garantindo uma experiência de uso rápida e eficiente. A infraestrutura de zonas de disponibilidade da AWS é projetada para ser altamente resiliente e segura. Cada zona de disponibilidade é construída com sistemas redundantes e equipamentos de última geração, garantindo a proteção dos dados e a confiabilidade dos serviços. Além disso, a AWS adota práticas de segurança rigorosas, como criptografia de dados, monitoramento avançado e controle de acesso, para garantir a proteção das informações dos usuários. Em resumo, a arquitetura de zonas de disponibilidade da AWS é um elemento-chave para a entrega de serviços em nuvem confiáveis, escaláveis e resilientes. Essa abordagem permite que os usuários tenham acesso a recursos de computação em nuvem com alta disponibilidade, tolerância a falhas e baixa latência em diversas regiões do mundo. Com essa infraestrutura sólida e global, a AWS se destaca como uma escolha popular para empresas que buscam aproveitar os benefícios da computação em nuvem em escala global.

4.1.1. AWS Rekognition



Amazon Rekognition

Figura 9 – Amazon Rekognition (Fonte: <https://aws.amazon.com/pt/>)

AWS Rekognition é um serviço de visão computacional fornecido pela AWS que permite a análise e reconhecimento de imagens e vídeos. Ele utiliza tecnologias de inteligência artificial e aprendizado de máquina para identificar objetos, pessoas, texto e conteúdo inapropriado em imagens e vídeos.

O serviço permite o reconhecimento de faces em fotos e vídeos, além de fornecer informações como emoções, idade, gênero e outras características. Ele também permite a busca de imagens por conteúdo, ou seja, é possível fazer pesquisas por imagens que contenham objetos específicos, como um carro ou um animal de estimação.

Além disso, o AWS Rekognition possui recursos avançados como detecção de celebridades em imagens, análise de expressões faciais e reconhecimento de atividades em vídeos. O serviço pode ser integrado facilmente a outras ferramentas da AWS, como o Amazon S3 para armazenamento de imagens e vídeos, e o Amazon Kinesis para processamento de fluxos de dados em tempo real.

O AWS Rekognition é utilizado em diversos casos de uso, incluindo segurança e vigilância, marketing e publicidade, entretenimento e educação. Por exemplo, empresas de segurança podem utilizar o serviço para monitorar câmeras de segurança, é possível usar o Rekognition para monitorar as câmeras de segurança de um shopping center e identificar atividades suspeitas, enquanto empresas de marketing podem usar o serviço para analisar as preferências dos clientes com base em suas interações com imagens e vídeos.

O serviço é totalmente gerenciado e escalável, o que significa que os usuários não precisam se preocupar com a infraestrutura subjacente e podem escalar facilmente seus recursos de acordo com suas necessidades. O AWS Rekognition é fácil de usar e possui uma API bem documentada, o que permite a integração com outras aplicações e serviços. Além disso, ele é totalmente gerenciado e escalável, o que significa que os usuários não precisam se preocupar com a infraestrutura subjacente e podem aumentar ou diminuir seus recursos de acordo com suas necessidades. O AWS Rekognition é uma solução poderosa e eficiente para análise de imagens e vídeos, e pode ser uma ferramenta valiosa para empresas de todos os tamanhos e em diversas áreas de atuação.

5. Capítulo 3: Projeto de Análise de fotos com AWS Rekognition

O presente capítulo descreve o projeto de análise de fotos utilizando a ferramenta Rekognition da AWS. Serão apresentados os processos envolvidos na configuração da infraestrutura necessária no ambiente da AWS e o desenvolvimento do código.

A análise de fotos é uma tarefa cada vez mais comum em diversas áreas, e a utilização de ferramentas de inteligência artificial, como o AWS Rekognition, pode trazer ganhos significativos em termos de eficiência e precisão na análise.

5.1. AWS e Credenciais

Foi necessário criar uma conta do zero na AWS, é uma conta de nível gratuito para uso pessoal com suporte básico, é necessário confirmar endereço

e cartão de crédito apenas para efeito de cadastro. Nessa conta, conseguimos utilizar o Rekognition e suas funcionalidades por até 1 ano.


Cadastrar-se na AWS

Selecionar um plano de suporte

Escolha um plano de suporte para sua conta comercial ou pessoal. [Compare planos e exemplos de definição de preço](#). Você pode alterar seu plano a qualquer momento no Console de Gerenciamento da AWS.


☒ **Suporte Basic - gratuito**

- Recomendado para novos usuários que estão começando a usar a AWS
- Acesso por autoatendimento 24 horas por dia/7 dias por semana a todos os recursos da AWS
- Apenas para questões de conta e faturamento
- Acesso ao Personal Health Dashboard e ao Trusted Advisor




☐ **Suporte Developer - a partir de 29 USD/mês**


- Recomendado para desenvolvedores que estão avaliando a AWS
- Acesso por e-mail ao AWS Support durante o horário comercial
- Tempo de resposta de 12 horas (horário comercial)



☐ **Suporte Business - a partir de 100 USD/mês**

- Recomendado para executar cargas de trabalho de produção na AWS
- Suporte técnico 24 horas por dia/7 dias por semana por e-mail, telefone e chat
- Tempo de resposta de 1 hora
- Conjunto completo de recomendações de melhores práticas do Trusted Advisor



**Precisa de suporte de nível Enterprise?**

A partir de 15.000 USD por mês, você receberá tempo de resposta de 15 minutos e experiência ao estilo de concierge com um Gerente técnico da conta designado. [Saiba mais](#)

Concluir cadastramento

Figura 10 – Tela de Cadastro da AWS

Após o cadastro feito, temos que gerar credencial (um arquivo CSV). Essa credencial serve de login para a AWS e suas APIs dentro do programa em Python.

5.2. Bibliotecas necessárias no código

É necessário o uso de 3 bibliotecas para python nesse código para a sua execução:

```
import csv
# biblioteca que se conecta com a AWS
import boto3
import openpyxl
```

Figura 11 – Bibliotecas usadas no código

CSV: biblioteca para ler e escrever arquivos CSV, manipular dados em tabelas, realizar operações de limpeza e processamento de dados e até mesmo exportar resultados em diferentes formatos de arquivo.

Boto3: biblioteca fornecida pela AWS para interagir com os serviços da AWS por meio de APIs, com essa biblioteca é possível criar, configurar e gerenciar recursos da AWS diretamente de um código Python. Para usar essa biblioteca é necessário instalá-la por meio do gerenciador de pacotes pip.

Openpyxl: É uma biblioteca Python que permite trabalhar com arquivos Excel em formato xlsx. Ele oferece uma interface fácil de usar para criar, ler e editar arquivos Excel, além de suportar vários recursos avançados, como gráficos, fórmulas, estilos, formatação condicional e muito mais.

5.3. Criação do cliente da AWS no código

```
# Cria um cliente da AWS lendo as credenciais do arquivo credentials.csv
with open('C:/Users/ezequ/Desktop/UNICARIOCA/1TCC/AWS/credentials.csv', 'r') as input:
    next(input)
    reader = csv.reader(input)
    for line in reader:
        access_key_id = line[2]
        secret_access_key = line[3]
```

Figura 12 – Parte do código onde o cliente da AWS é criado localmente

Essa linha de código cria um cliente da AWS (Amazon Web Services) lendo as credenciais do arquivo "credentials.csv" gerado anteriormente no site da AWS. O arquivo "credentials.csv" deve estar localizado no caminho "C:/Users/ezequ/Desktop/UNICARIOCA/1TCC/AWS/". O código utiliza a biblioteca csv do Python para ler o arquivo "credentials.csv". A primeira linha do arquivo é ignorada (usando o comando `next(input)`) porque é assumido que a primeira linha contém o cabeçalho do arquivo, e não as informações das credenciais. Em seguida, o arquivo é lido linha por linha por meio do objeto `reader`, que é criado a partir do arquivo "input" e para cada linha lida, as informações das credenciais são extraídas e armazenadas nas variáveis "access_key_id" e "secret_access_key". Essas informações de credenciais

serão usadas posteriormente para autenticar o cliente da AWS e permitir que ele acesse os serviços da AWS.

5.4. Acessando o rekognition no código

```
# Cria o cliente de acesso a API Rekognition da AWS
client = boto3.client('rekognition',
                      aws_access_key_id=access_key_id,
                      aws_secret_access_key=secret_access_key,
                      region_name='us-east-1')
```

Figura 13 – Parte do código onde o AWS Rekognition é chamado

Essas linhas de código são usadas para criar um cliente para acessar o serviço de reconhecimento de imagens da Amazon Web Services (AWS), chamado "Rekognition". A biblioteca boto3 do Python é usada para interagir com os serviços da AWS. O código cria um objeto "client" da classe boto3.client, que é usado para fazer solicitações ao serviço Rekognition. A função boto3.client aceita vários parâmetros, incluindo o nome do serviço (no caso, "rekognition"), as credenciais de autenticação da AWS (access_key_id e secret_access_key) e a região geográfica onde o serviço está hospedado (no caso, 'us-east-1').

O parâmetro "aws_access_key_id" é a chave de acesso da AWS que é usada para autenticar o cliente, enquanto o "aws_secret_access_key" é a chave secreta correspondente. Essas informações de autenticação são as mesmas que foram lidas do arquivo "credentials.csv" no exemplo anterior. Depois de criar o objeto "client", é possível usar seus métodos para enviar solicitações para o serviço Rekognition e receber respostas do serviço. Por exemplo, o método "detect_labels" pode ser usado para identificar objetos e rótulos em uma imagem. O uso desses métodos dependerá do objetivo específico do TCC e do uso do serviço de reconhecimento de imagens da AWS.

5.5. Seleção da foto e conversão

```
# Usa a foto
photo = "C:/Users/ezequ/Desktop/UNICARIOCA/1TCC/AWS/rio.jfif"

# Converte uma imagem em um array de bytes
with open(photo, 'rb') as source_image:
    source_bytes = source_image.read()
```

Figura 14 – Parte do código onde é possível escolher uma foto para análise

A primeira linha define o caminho que contém a imagem, que pode ser em vários formatos, entre eles: jpg, jpeg, jfif, png, bmp.

A segunda linha usa a declaração "with open()" para abrir o arquivo de imagem em modo binário ('rb') e atribui-lo a uma variável "source_image". Em seguida, o método ".read()" é usado para ler todo o conteúdo do arquivo em uma variável "source_bytes".

Esse processo é necessário porque o serviço de reconhecimento de imagens da AWS espera que a imagem esteja na forma de um array de bytes, e não em um arquivo de imagem convencional. Essa conversão de imagem para um array de bytes ocorre da seguinte forma: A conversão envolve a transformação dos valores dos pixels da imagem em uma série de bytes que podem ser interpretados por um computador.

O processo de conversão começa com a leitura da imagem em seu formato original. Em seguida, cada pixel na imagem é examinado e seu valor é convertido em um ou mais bytes, dependendo da profundidade de bits da imagem. Por exemplo, uma imagem de 8 bits terá um byte para cada pixel, enquanto uma imagem de 24 bits terá três bytes para cada pixel.

Os valores dos bytes resultantes são geralmente organizados em uma matriz (array), onde cada elemento representa um byte da imagem. A ordem dos bytes pode variar dependendo do formato da imagem e da plataforma em que ela está sendo usada. Uma vez que a imagem tenha sido convertida em uma array de bytes, ela pode ser armazenada em um arquivo ou transmitida através

de uma rede. Quando a imagem precisa ser acessada novamente, ela pode ser lida a partir do arquivo ou da rede e convertida de volta em sua forma original de imagem.

É importante notar que a conversão de imagens para uma array de bytes pode resultar em perda de informações, especialmente quando ocorre compressão de imagem. Além disso, o tamanho da array de bytes resultante pode ser muito grande, o que pode afetar a eficiência do armazenamento e da transmissão de imagens em grandes quantidades.

5.6. Camadas reconhecidas nas fotos, configuração da confiança e verificação de Rostos

```
# Usa o cliente para detectar as labels (camadas, o quão longe ele pode se aprofundar no reconhecimento)
response = client.detect_labels(Image={'Bytes': source_bytes},
                                MaxLabels=10,
                                MinConfidence=90)

# Verifica se há rostos na foto
face_response = client.detect_faces(Image={'Bytes': source_bytes}, Attributes=["ALL"])
has_face = False
if len(face_response["FaceDetails"]) > 0:
    has_face = True
    print("Foram encontrados rostos na foto.")
else:
    print("Não foram encontradas rostos na foto.")
```

Figura 15 – Parte do código onde é possível configurar o Rekognition

Este trecho de código faz uso do AWS Rekognition e é onde podemos configurar a partir de 3 argumentos dentro da função "detect_labels()":

"Image" - um dicionário que especifica a imagem que será processada. O valor da chave 'Bytes' deve ser uma sequência de bytes que representa a imagem a ser analisada.

"MaxLabels" - um valor inteiro que podemos configurar que indica o número máximo de labels que o algoritmo pode retornar para essa imagem. Neste caso, a quantidade máxima é de 10 labels.

"MinConfidence" - um valor float que podemos configurar para indicar a confiança mínima necessária para que uma label seja retornada como resultado da análise. Neste caso, somente as labels cuja confiança é superior a 90% serão retornadas.

O objeto "response" armazena o resultado da análise da imagem realizada pelo algoritmo de reconhecimento de imagem. Esse resultado é uma lista de dicionários, onde cada dicionário contém informações sobre uma label detectada na imagem. As informações incluem o nome da label e a confiança atribuída pelo algoritmo a essa label.

Já para a verificação de rostos, o código inicia com a criação de uma variável "face_response", que armazena a resposta do serviço da AWS após a detecção de rostos na imagem fornecida. A variável "Attributes" define quais atributos devem ser retornados na resposta, nesse caso, todos ("ALL"). Em seguida, uma variável "has_face" é definida como falsa por padrão, indicando que inicialmente não foi encontrado nenhum rosto na imagem. O código verifica se há detalhes de rostos na resposta retornada pela biblioteca do AWS Rekognition, acessando a chave "FaceDetails" do dicionário "face_response". Se houver detalhes de rostos, a variável "has_face" é definida como verdadeira e é exibida uma mensagem informando que rostos foram encontrados na imagem. Caso contrário, a mensagem indicando que não foram encontradas faces é exibida.

5.7. Armazenamento dos resultados

```
# Tenta abrir o arquivo .xlsx, cria um novo se não existir
try:
    workbook = openpyxl.load_workbook("resultado.xlsx")
    worksheet = workbook.active
except FileNotFoundError:
    workbook = openpyxl.Workbook()
    worksheet = workbook.active
    # Escreve os cabeçalhos das colunas
    worksheet.cell(row=1, column=1, value="Label (Rótulo)")
    worksheet.cell(row=1, column=2, value="Confidence (Confiança) em %")
    worksheet.cell(row=1, column=3, value="Detecção de rosto")
    worksheet.cell(row=1, column=4, value="Foto")

# Encontra a última linha com dados na planilha
last_row = worksheet.max_row

row = worksheet.max_row + 1
# Adiciona uma nova linha para cada label detectada
for i, label in enumerate(response["Labels"]):
    worksheet.cell(row=last_row + i + 1, column=1, value=label["Name"])
    worksheet.cell(row=last_row + i + 1, column=2, value=label["Confidence"])
    worksheet.cell(row=row, column=3, value=has_face)
    photo = photo.split("/")[-1]
    worksheet.cell(row=last_row + i + 1, column=4, value=photo)
    row += 1

# Salva o arquivo .xlsx
print("Salvando o arquivo...")
workbook.save("resultado.xlsx")
print("Arquivo salvo com sucesso!")
```

Figura 16 – Parte do código onde os resultados são exportados para um arquivo .xlsx

Os resultados obtidos através das análises são armazenados em um arquivo .xlsx para poderem ser consultados posteriormente, nesse arquivo temos as informações de Rótulos encontrados, a confiança que o Rekognition teve para encontrar esse rótulo, se houve detecção de rosto na foto e de qual foto foi feita a análise.

Primeiro, o código tenta abrir o arquivo "resultado.xlsx" com a função `openpyxl.load_workbook()`. Se o arquivo não existir, a exceção `FileNotFoundError` é capturada e um novo arquivo é criado com `openpyxl.Workbook()`. Em seguida, os cabeçalhos das colunas são escritos na primeira linha da planilha com `worksheet.cell()`. Depois, o código encontra a última linha com dados na planilha com `worksheet.max_row` e cria uma linha com `row = worksheet.max_row + 1`. Para cada label detectada, o código adiciona uma nova linha na planilha com `worksheet.cell()`, preenchendo as colunas com os valores do label, sua confiança, a detecção de rosto (`has_face`) e o nome da foto (`photo`). A variável `row` é incrementada para garantir que cada linha tenha um número diferente. Por fim, o código salva o arquivo .xlsx com a função `workbook.save()` e exibe uma mensagem informando que o arquivo foi salvo com sucesso. Esse processo é útil para armazenar os resultados de uma aplicação em um formato que possa ser facilmente compartilhado e visualizado.

5.8. Execução do código

Existem várias maneiras de executar um código em Python dentro do VS Code, para esse código foram utilizadas duas maneiras:

-Utilizando o depurador do VS Code:

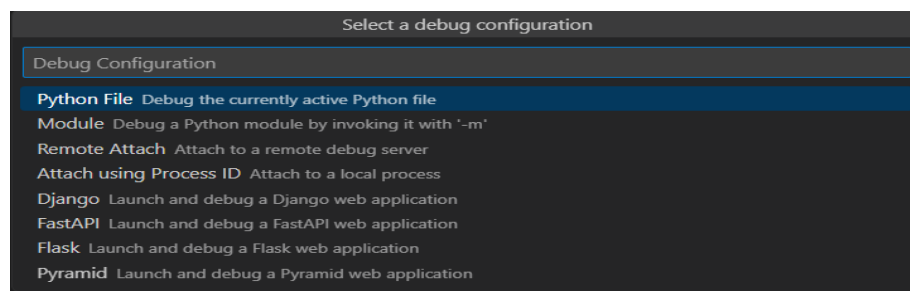


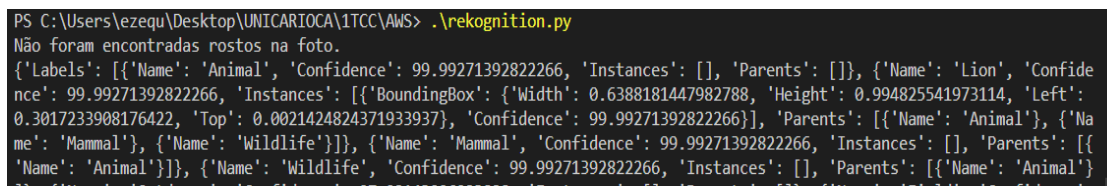
Figura 17 – Depurador nativo do VS Code

Essa é uma ferramenta útil para detectar erros e problemas em seu código Python principalmente na fase de construção de um código. Com ele também podemos inspecionar o estado atual das variáveis e executar o código linha por linha. Para usá-lo, você pode colocar pontos de interrupção em seu código e, em seguida, executar o depurador pressionando F5 ou usando a paleta de comandos para "Run Python File in Debugger". O depurador irá pausar a execução do código no ponto de interrupção.

- Primeira fase-Colocar pontos de interrupção no código: Para usar o depurador, é preciso definir pontos de interrupção em seu código. Isso pode ser feito clicando no número da linha em que deseja colocar um ponto de interrupção ou pressionando F9 enquanto estiver na linha. O ponto de interrupção será exibido como uma bola vermelha.
- Segunda fase-Iniciar o depurador: Depois de definir os pontos de interrupção, abra o arquivo Python que deseja depurar e clique no botão "Run and Debug" na barra lateral ou pressione F5. Isso abrirá a janela de depuração e será solicitado a escolha do ambiente de execução. Selecione o interpretador Python correto e escolha "Python File" para depurar o arquivo Python.
- Terceira fase- Executar o código em modo de depuração: Agora, o código está pronto para ser executado em modo de depuração. Quando o programa chegar ao ponto de interrupção, a execução será pausada e assim será possível ver o estado atual das variáveis. Os controles na barra superior da janela de depuração são usados para continuar a execução passo a passo, para executar até o próximo ponto de interrupção ou para finalizar a execução.
- Quarta fase- Inspecionar o estado das variáveis: Enquanto o programa está em pausa, é possível inspecionar o valor atual das variáveis. Também é possível adicionar variáveis para monitorar no painel de depuração clicando em "Add Expression" e digitando o nome da variável. O valor da variável será atualizado à medida que o programa é executado.

- Quinta fase-Correção de erros: Se o programa falhar em algum ponto, o depurador exibirá uma mensagem de erro. É possível usar as informações fornecidas na mensagem de erro para corrigir o problema em seu código e executar novamente o programa em modo de depuração.
- Sexta fase-Conclua a execução: Quando a execução do programa estiver concluída, o depurador será encerrado e será possível ver o resultado no terminal integrado ou em uma janela separada.

-Utilizando o Terminal integrado do VS Code:



```
PS C:\Users\ezequ\Desktop\UNICARIOCA\1TCC\AWS> .\rekognition.py
Não foram encontradas rostos na foto.
{"Labels": [{"Name": "Animal", "Confidence": 99.99271392822266, "Instances": [], "Parents": []}, {"Name": "Lion", "Confidence": 99.99271392822266, "Instances": [{"BoundingBox": {"Width": 0.6388181447982788, "Height": 0.994825541973114, "Left": 0.3017233908176422, "Top": 0.0021424824371933937}, "Confidence": 99.99271392822266}], "Parents": [{"Name": "Animal"}, {"Name": "Mammal"}, {"Name": "Wildlife"}]}, {"Name": "Mammal", "Confidence": 99.99271392822266, "Instances": [], "Parents": [{"Name": "Animal"}, {"Name": "Wildlife"}]}, {"Name": "Wildlife", "Confidence": 99.99271392822266, "Instances": [], "Parents": [{"Name": "Animal"}]}
```

Figura 18 – Terminal integrado do VS Code Executando o programa

Esse método de execução é usado logo após ter feito a depuração do código e verificado que está tudo estável. Para rodar o arquivo no terminal é preciso verificar se o terminal já está na pasta do arquivo, caso não esteja, é necessário navegar por entre as pastas utilizando o comando 'cd' até chegar na pasta onde o arquivo com o código está armazenado. Depois de estar na pasta do arquivo, basta digitar o nome do arquivo que você deseja executar com o final '.py' para indicar que este é um arquivo com código Python.

6. Capítulo 4: Resultados

Neste capítulo, serão apresentados os principais resultados alcançados com o programa desenvolvido, incluindo testes realizados, métricas utilizadas e análises feitas. Vale ressaltar que a análise de rostos presentes na foto, é uma função independente das camadas, é um outro recurso do Rekognition. Por exemplo, o programa pode falar que não tem rostos na foto, mas em alguma camada pode ser que venha algo relacionado a rostos ou pessoas.

6.1. Usando retratos, fotos com pessoas visíveis:

6.1.1. Foto 1, uma 3x4, até 10 camadas e o mínimo de confiança em 90%



Figura 19 – Foto Com rosto 1

- Saída do programa:

```
Foram encontrados rostos na foto.
{'Labels': [{'Name': 'Face', 'Confidence': 99.99766540527344, 'Instances': [], 'Parents': [{'Name': 'Head'}, {'Name': 'Person'}]}, {'Name': 'Head', 'Confidence': 99.99766540527344, 'Instances': [], 'Parents': [{'Name': 'Person'}]}, {'Name': 'Neck', 'Confidence': 99.99766540527344, 'Instances': [], 'Parents': [{'Name': 'Body Part'}, {'Name': 'Face'}, {'Name': 'Head'}, {'Name': 'Person'}]}, {'Name': 'Person', 'Confidence': 99.99766540527344, 'Instances': [{'BoundingBox': {'Width': 0.8765063881874084, 'Height': 0.9928143620491028, 'Left': 0.07294944673776627, 'Top': 0.004122306127101183}, 'Confidence': 99.73159790039062}], 'Parents': []}, {'Name': 'Photography', 'Confidence': 99.99620056152344, 'Instances': [], 'Parents': []}, {'Name': 'Portrait', 'Confidence': 99.99620056152344, 'Instances': [], 'Parents': [{'Name': 'Face'}, {'Name': 'Head'}, {'Name': 'Person'}, {'Name': 'Photography'}]}, {'Name': 'T-Shirt', 'Confidence': 99.88600158691406, 'Instances': [], 'Parents': [{'Name': 'Clothing'}]}, {'Name': 'Adult', 'Confidence': 99.73159790039062, 'Instances': [{'BoundingBox': {'Width': 0.8765063881874084, 'Height': 0.9928143620491028, 'Left': 0.07294944673776627, 'Top': 0.004122306127101183}, 'Confidence': 99.73159790039062}], 'Parents': [{'Name': 'Person'}]}, {'Name': 'Male', 'Confidence': 99.73159790039062, 'Instances': [{'BoundingBox': {'Width': 0.8765063881874084, 'Height': 0.9928143620491028, 'Left': 0.07294944673776627, 'Top': 0.004122306127101183}, 'Confidence': 99.73159790039062}], 'Parents': [{'Name': 'Adult'}, {'Name': 'Male'}, {'Name': 'Person'}]}], 'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': 'e52a7306-4383-4768-9876-0f42b6c88897', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': 'e52a7306-4383-4768-9876-0f42b6c88897', 'content-type': 'application/x-amz-json-1.1', 'content-length': '2350', 'date': 'Sun, 07 May 2023 14:49:51 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 20 – Resultado da Figura 19 no terminal

- Saída do Excel:

Label (Rótulo)	Confidence (Confiança) em %	Detecção de rosto	Foto
Face	99,99766541	VERDADEIRO	foto3x4.jpg
Head	99,99766541	VERDADEIRO	foto3x4.jpg
Neck	99,99766541	VERDADEIRO	foto3x4.jpg
Person	99,99766541	VERDADEIRO	foto3x4.jpg
Photography	99,99620056	VERDADEIRO	foto3x4.jpg
Portrait	99,99620056	VERDADEIRO	foto3x4.jpg
T-Shirt	99,88600159	VERDADEIRO	foto3x4.jpg
Adult	99,7315979	VERDADEIRO	foto3x4.jpg
Male	99,7315979	VERDADEIRO	foto3x4.jpg
Man	99,7315979	VERDADEIRO	foto3x4.jpg

Figura 21 – Resultado da Figura 19 no Excel

6.1.2. Foto 2, usando o piloto de Fórmula 1, Lewis Hamilton como exemplo, até 10 camadas e o mínimo de confiança em 90%



Figura 22 – Foto com rosto 2

- Saída do programa:

```
Foram encontrados rostos na foto.
{'Labels': [{'Name': 'Person', 'Confidence': 99.69581604003906, 'Instances': [{'BoundingBox': {'Width': 0.9727132320404053, 'Height': 0.9807943105697632, 'Left': 0.020550040528178215, 'Top': 0.019192786887288094}, 'Confidence': 99.58924102783203}], 'Parents': [], {'Name': 'Solo Performance', 'Confidence': 99.69581604003906, 'Instances': [], 'Parents': [{'Name': 'Performer'}, {'Name': 'Person'}]}, {'Name': 'Adult', 'Confidence': 99.58924102783203, 'Instances': [{'BoundingBox': {'Width': 0.9727132320404053, 'Height': 0.9807943105697632, 'Left': 0.020550040528178215, 'Top': 0.019192786887288094}, 'Confidence': 99.58924102783203}], 'Parents': [{'Name': 'Person'}]}, {'Name': 'Male', 'Confidence': 99.58924102783203, 'Instances': [{'BoundingBox': {'Width': 0.9727132320404053, 'Height': 0.9807943105697632, 'Left': 0.020550040528178215, 'Top': 0.019192786887288094}, 'Confidence': 99.58924102783203}], 'Parents': [{'Name': 'Person'}]}, {'Name': 'Man', 'Confidence': 99.58924102783203, 'Instances': [{'BoundingBox': {'Width': 0.9727132320404053, 'Height': 0.9807943105697632, 'Left': 0.020550040528178215, 'Top': 0.019192786887288094}, 'Confidence': 99.58924102783203}], 'Parents': [{'Name': 'Adult'}, {'Name': 'Male'}, {'Name': 'Person'}]}, {'Name': 'Head', 'Confidence': 99.14372253417969, 'Instances': [], 'Parents': [{'Name': 'Person'}]}, {'Name': 'Face', 'Confidence': 99.1372229003906, 'Instances': [], 'Parents': [{'Name': 'Head'}, {'Name': 'Person'}]}, {'Name': 'Dimples', 'Confidence': 98.42189025878906, 'Instances': [], 'Parents': [{'Name': 'Face'}, {'Name': 'Head'}, {'Name': 'Person'}]}, {'Name': 'Electrical Device', 'Confidence': 96.33438110351562, 'Instances': [], 'Parents': [], {'Name': 'Microphone', 'Confidence': 96.33438110351562, 'Instances': [{'BoundingBox': {'Width': 0.1749751716852188, 'Height': 0.2939321994781494, 'Left': 0.2170657217502594, 'Top': 0.3574431538581848}, 'Confidence': 96.33438110351562}], 'Parents': [{'Name': 'Electrical Device'}]}], 'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': '30dfef78-c370-4fa3-84cc-1c9a9bbcb860', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': '30dfef78-c370-4fa3-84cc-1c9a9bbcb860', 'content-type': 'application/x-amz-json-1.1', 'content-length': '2464', 'date': 'Sun, 07 May 2023 15:11:42 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 23 – Resultado da Figura 22 no terminal

- Saída do Excel:

Label (Rótulo)	Confidence	Deteção de rosto	Foto
Person	99,69582	VERDADEIRO	hamilton.jpg
Solo Performance	99,69582	VERDADEIRO	hamilton.jpg
Adult	99,58924	VERDADEIRO	hamilton.jpg
Male	99,58924	VERDADEIRO	hamilton.jpg
Man	99,58924	VERDADEIRO	hamilton.jpg
Head	99,14372	VERDADEIRO	hamilton.jpg
Face	99,13722	VERDADEIRO	hamilton.jpg
Dimples	98,42189	VERDADEIRO	hamilton.jpg
Electrical Device	96,33438	VERDADEIRO	hamilton.jpg
Microphone	96,33438	VERDADEIRO	hamilton.jpg

Figura 24 – Resultado da Figura 22 no Excel

6.1.3. Foto 3, apenas um rosto visível, até 10 camadas e o mínimo de confiança em 90%



Figura 25 – Foto com rosto 3

- Saída do programa:

```
Foram encontrados rostos na foto.
{'Labels': [{'Name': 'Clothing', 'Confidence': 100.0, 'Instances': [], 'Parents': []}, {'Name': 'T-Shirt', 'Confidence': 100.0, 'Instances': [], 'Parents': [{'Name': 'Clothing'}]}, {'Name': 'Face', 'Confidence': 99.86861419677734, 'Instances': [], 'Parents': [{'Name': 'Head'}, {'Name': 'Person'}]}, {'Name': 'Head', 'Confidence': 99.86861419677734, 'Instances': [], 'Parents': [{'Name': 'Person'}]}, {'Name': 'Person', 'Confidence': 99.86861419677734, 'Instances': [{'BoundingBox': {'Width': 1.0, 'Height': 0.693555934906006, 'Left': 0.0, 'Top': 0.3063969612121582}, 'Confidence': 98.96273803710938]}, 'Parents': []}, {'Name': 'Photography', 'Confidence': 99.86861419677734, 'Instances': [], 'Parents': []}, {'Name': 'Portrait', 'Confidence': 99.86861419677734, 'Instances': [], 'Parents': [{'Name': 'Face'}, {'Name': 'Head'}, {'Name': 'Person'}, {'Name': 'Photography'}]}, {'Name': 'Accessories', 'Confidence': 91.30274963378906, 'Instances': [], 'Parents': []}], 'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': 'e6b591b4-2ce0-43e8-9b1b-b8e07d04edf6', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': 'e6b591b4-2ce0-43e8-9b1b-b8e07d04edf6', 'content-type': 'application/x-amz-json-1.1', 'content-length': '1461', 'date': 'Mon, 01 May 2023 17:03:03 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
```

Figura 26 – Resultado da Figura 25 no terminal

- Saída do arquivo excel:

Label (Rótulo)	Confidenc	Detecção de rosto	Foto
Clothing	100	VERDADEIRO	rosto1.jpg
T-Shirt	100	VERDADEIRO	rosto1.jpg
Face	99,86861	VERDADEIRO	rosto1.jpg
Head	99,86861	VERDADEIRO	rosto1.jpg
Person	99,86861	VERDADEIRO	rosto1.jpg
Photography	99,86861	VERDADEIRO	rosto1.jpg
Portrait	99,86861	VERDADEIRO	rosto1.jpg
Accessories	91,30275	VERDADEIRO	rosto1.jpg

Figura 27 – Resultado da Figura 25 no Excel

6.1.4. Foto 4, 3 rostos visíveis, até 10 camadas e o mínimo de confiança em 90%



Figura 27 – Foto com 3 rostos

- Saída do programa:

```
Foram encontrados rostos na foto.
{'Labels': [{'Name': 'People', 'Confidence': 99.999771118164, 'Instances': [], 'Parents': [{'Name': 'Person'}]}, {'Name': 'Person', 'Confidence': 99.999771118164, 'Instances': [{'BoundingBox': {'Width': 0.28833746910095215, 'Height': 0.5125454664230347, 'Left': 0.0004812908300664276, 'Top': 0.48741477727890015}, 'Confidence': 98.99047088623047}, {'BoundingBox': {'Width': 0.6001436114311218, 'Height': 0.5155636072158813, 'Left': 0.13249951601028442, 'Top': 0.4835675358772278}, 'Confidence': 98.8919906616211}, {'BoundingBox': {'Width': 0.39768755435943604, 'Height': 0.4910009801387787, 'Left': 0.6013712286949158, 'Top': 0.5089990496635437}, 'Confidence': 98.3870391845703}], {'Name': 'Baseball Cap', 'Confidence': 99.99989318847656, 'Instances': [], 'Parents': [{'Name': 'Cap'}, {'Name': 'Clothing'}, {'Name': 'Hat'}]}, {'Name': 'Cap', 'Confidence': 99.99989318847656, 'Instances': [], 'Parents': [{'Name': 'Clothing'}, {'Name': 'Hat'}]}, {'Name': 'Hat', 'Confidence': 99.99989318847656, 'Instances': [], 'Parents': []}], {'Name': 'T-Shirt', 'Confidence': 99.9948959350586, 'Instances': [], 'Parents': [{'Name': 'Clothing'}]}, {'Name': 'Child', 'Confidence': 98.99047088623047, 'Instances': [{'BoundingBox': {'Width': 0.28833746910095215, 'Height': 0.5125454664230347, 'Left': 0.0004812908300664276, 'Top': 0.48741477727890015}, 'Confidence': 98.99047088623047}], 'Parents': [{'Name': 'Person'}]}, {'Name': 'Female', 'Confidence': 98.99047088623047, 'Instances': [{'BoundingBox': {'Width': 0.28833746910095215, 'Height': 0.5125454664230347, 'Left': 0.0004812908300664276, 'Top': 0.48741477727890015}, 'Confidence': 98.99047088623047}], 'Parents': [{'Name': 'Person'}]}, {'Name': 'Girl', 'Confidence': 98.99047088623047, 'Instances': [{'BoundingBox': {'Width': 0.28833746910095215, 'Height': 0.5125454664230347, 'Left': 0.0004812908300664276, 'Top': 0.48741477727890015}, 'Confidence': 98.99047088623047}], 'Parents': [{'Name': 'Person'}]}, {'Name': 'Teen', 'Confidence': 98.8919906616211, 'Instances': [{'BoundingBox': {'Width': 0.6001436114311218, 'Height': 0.5155636072158813, 'Left': 0.13249951601028442, 'Top': 0.4835675358772278}, 'Confidence': 98.8919906616211}], 'Parents': [{'Name': 'Person'}]}, {'Name': 'Person', 'Confidence': 98.8919906616211, 'Instances': [], 'Parents': []}], 'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': '68dbc0eb-a8e3-4d10-a81d-d06ec5039c51', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': '68dbc0eb-a8e3-4d10-a81d-d06ec5039c51', 'content-type': 'application/x-amz-json-1.1', 'content-length': '3217', 'date': 'Mon, 01 May 2023 17:21:10 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 28 – Resultado da Figura 27 no terminal

- Saída do arquivo Excel:

Label (Rótulo)	Confidence (Cor)	Deteção de rosto	Foto
People	99,99997711	VERDADEIRO	rosto2.jpg
Person	99,99997711	VERDADEIRO	rosto2.jpg
Baseball Cap	99,99989319	VERDADEIRO	rosto2.jpg
Cap	99,99989319	VERDADEIRO	rosto2.jpg
Hat	99,99989319	VERDADEIRO	rosto2.jpg
T-Shirt	99,99489594	VERDADEIRO	rosto2.jpg
Child	98,99047089	VERDADEIRO	rosto2.jpg
Female	98,99047089	VERDADEIRO	rosto2.jpg
Girl	98,99047089	VERDADEIRO	rosto2.jpg
Teen	98,89199066	VERDADEIRO	rosto2.jpg

Figura 29 – Resultado da Figura 27 no Excel

E como o programa reage com fotos de menor resolução?

- 6.2.** Foto 5, 3 rostos não muito visíveis devido à baixa resolução da foto, até 10 camadas e o mínimo de confiança em 90%



Figura 30 – Foto de rua feita por celular

- Saída do programa:

```
Não foram encontradas rostos na foto.
{'Labels': [{'Name': 'Car', 'Confidence': 99.74601745605469, 'Instances': [{'BoundingBox': {'Width': 0.5753121972084045, 'Height': 0.34526771306991577, 'Left': 0.09040767699480057, 'Top': 0.3654140830039978}, 'Confidence': 99.74601745605469}, {'BoundingBox': {'Width': 0.26896706223487854, 'Height': 0.42118197679519653, 'Left': 0.7287542819976807, 'Top': 0.5780863165855408}, 'Confidence': 96.95150756835938}], 'Parents': [{'Name': 'Transportation'}, {'Name': 'Vehicle'}]}, {'Name': 'Transportation', 'Confidence': 99.74601745605469, 'Instances': [], 'Parents': []}, {'Name': 'Vehicle', 'Confidence': 99.74601745605469, 'Instances': [], 'Parents': [{'Name': 'Transportation'}]}, {'Name': 'Person', 'Confidence': 97.53487396240234, 'Instances': [{'BoundingBox': {'Width': 0.09055694192647934, 'Height': 0.22943833470344543, 'Left': 0.0, 'Top': 0.37687793374061584}, 'Confidence': 97.53487396240234}, {'BoundingBox': {'Width': 0.037926673889160156, 'Height': 0.062741219997406, 'Left': 0.6532337069511414, 'Top': 0.3488486409187317}, 'Confidence': 90.9023208618164}], 'Parents': []}], 'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': '6e900332-0b4f-4045-9809-ddbb5f2fdbad', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': '6e900332-0b4f-4045-9809-ddbb5f2fdbad', 'content-type': 'application/x-amz-json-1.1', 'content-length': '1315', 'date': 'Mon, 01 May 2023 17:35:03 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 31 – Resultado da Figura 30 no terminal

- Saída do arquivo Excel:

Label (Rótulo)	Confidence	Deteção de rosto	Foto
Car	99,74602	FALSO	fusca.jpeg
Transportation	99,74602	FALSO	fusca.jpeg
Vehicle	99,74602	FALSO	fusca.jpeg
Person	97,53487	FALSO	fusca.jpeg

Figura 32 – Resultado da Figura 30 no Excel

- Mantendo a quantidade de camadas e diminuindo a confiança para 75%:

```
Não foram encontradas rostos na foto.
{'Labels': [{'Name': 'Car', 'Confidence': 99.74601745605469, 'Instances': [{'BoundingBox': {'Width': 0.5753121972084045, 'Height': 0.34526771306991577, 'Left': 0.09040767699480057, 'Top': 0.3654140830039978}, {'Confidence': 99.74601745605469}, {'BoundingBox': {'Width': 0.26896706223487854, 'Height': 0.42118197679519653, 'Left': 0.7287542819976807, 'Top': 0.5780863165855408}, {'Confidence': 96.95150756835938}], 'Parents': [{'Name': 'Transportation'}, {'Name': 'Vehicle'}]}, {'Name': 'Transportation', 'Confidence': 99.74601745605469, 'Instances': [], 'Parents': []}, {'Name': 'Vehicle', 'Confidence': 99.74601745605469, 'Instances': [], 'Parents': []}, {'Name': 'Person', 'Confidence': 97.53487396240234, 'Instances': [{'BoundingBox': {'Width': 0.09055694192647934, 'Height': 0.22943833470344543, 'Left': 0.0, 'Top': 0.37687793374061584}, {'Confidence': 97.53487396240234}, {'BoundingBox': {'Width': 0.037926673889160156, 'Height': 0.062741219997406, 'Left': 0.6532337069511414, 'Top': 0.3488486409187317}, {'Confidence': 90.9023208618164}, {'BoundingBox': {'Width': 0.042769987136125565, 'Height': 0.07263293862342834, 'Left': 0.9451491236686707, 'Top': 0.34137094020843506}, {'Confidence': 87.24508666992188}], 'Parents': []}, {'Name': 'Machine', 'Confidence': 88.33523559570312, 'Instances': [], 'Parents': []}, {'Name': 'Wheel', 'Confidence': 88.33523559570312, 'Instances': [{'BoundingBox': {'Width': 0.08781669288873672, 'Height': 0.1363716721534729, 'Left': 0.2019605189561844, 'Top': 0.5737402439117432}, {'Confidence': 88.33523559570312}], 'Parents': []}, {'Name': 'Machine'}]}, {'Name': 'Urban', 'Confidence': 82.95955657958984, 'Instances': [], 'Parents': []}], 'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': '91035435-faf4-4e6b-b642-a390f0679a82', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-request-id': '91035435-faf4-4e6b-b642-a390f0679a82', 'content-type': 'application/x-amz-json-1.1', 'content-length': '2068', 'date': 'Mon, 01 May 2023 18:38:55 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 33 – Resultado da Figura 30 no terminal

Label (Rótulo)	Confidence	Deteção	Foto
Car	99,74602	FALSO	fusca.jpeg
Transportation	99,74602	FALSO	fusca.jpeg
Vehicle	99,74602	FALSO	fusca.jpeg
Person	97,53487	FALSO	fusca.jpeg
Machine	88,33524	FALSO	fusca.jpeg
Wheel	88,33524	FALSO	fusca.jpeg
Urban	82,95956	FALSO	fusca.jpeg

Figura 34 – Resultado da Figura 30 no Excel

O programa conseguiu reconhecer mais 3 camadas, mas duas delas ainda foram relacionadas ao carro azul que está mais em destaque na foto, o que mostra que em fotos de menor resolução, o programa tende a reconhecer menos elementos, como por exemplo, 3 pessoas um pouco mais distantes, mas com os rostos virados para a foto.

6.3. Testando o programa com fotos de Paisagem:

6.3.1. Dia de sol na praia, alguns rostos visíveis, até 10 camadas e o mínimo de confiança em 90%:



Figura 35 – Foto de paisagem 1

- Saída do Programa:

```
Foram encontrados rostos na foto.
{'Labels': [{'Name': 'City', 'Confidence': 99.88651275634766, 'Instances': [], 'Parents': []}, {'Name': 'Person', 'Confidence': 98.86178588867188, 'Instances': [{'BoundingBox': {'Width': 0.14121726155281067, 'Height': 0.12576188147068024, 'Left': 0.2513093054294586, 'Top': 0.8740750551223755}, 'Confidence': 98.86178588867188}, {'BoundingBox': {'Width': 0.11734267324209213, 'Height': 0.14871501922607422, 'Left': 0.15769264101982117, 'Top': 0.8512563705444336}, 'Confidence': 98.07856750488281}, {'BoundingBox': {'Width': 0.14820094406604767, 'Height': 0.08901171386241913, 'Left': 0.016447925940155983, 'Top': 0.9109882712364197}, 'Confidence': 97.48907470703125}, {'BoundingBox': {'Width': 0.05474152788519859, 'Height': 0.03555154800415039, 'Left': 0.4678875505924225, 'Top': 0.964213490486145}, 'Confidence': 96.29102325439453}, {'BoundingBox': {'Width': 0.06222598999738693, 'Height': 0.03927220404148102, 'Left': 0.2750089168548584, 'Top': 0.8538787961006165}, 'Confidence': 95.78546905517578}, {'BoundingBox': {'Width': 0.06400299072265625, 'Height': 0.07078774273395538, 'Left': 0.8368578553199768, 'Top': 0.8594892621040344}, 'Confidence': 94.22603607177734}, {'BoundingBox': {'Width': 0.020469779148697853, 'Height': 0.007808204740285873, 'Left': 0.792056143283844, 'Top': 0.8497799634933472}, 'Confidence': 91.74465942382812}], 'Parents': [], {'Name': 'Metropolis', 'Confidence': 97.59953308105469, 'Instances': [], 'Parents': [{'Name': 'City'}, {'Name': 'Urban'}]}, {'Name': 'Urban', 'Confidence': 97.59953308105469, 'Instances': [], 'Parents': []}, {'Name': 'Clothing', 'Confidence': 97.34571838378906, 'Instances': [], 'Parents': []}, {'Name': 'Hat', 'Confidence': 97.34571838378906, 'Instances': [{'BoundingBox': {'Width': 0.0805388018488884, 'Height': 0.028174882754683495, 'Left': 0.1910678744316101, 'Top': 0.8505117893218994}, 'Confidence': 97.34571838378906}], 'Parents': [{'Name': 'Clothing'}]}, {'Name': 'Flag', 'Confidence': 94.0333282470703, 'Instances': [{'BoundingBox': {'Width': 0.14565040171146393, 'Height': 0.22848518192768097, 'Left': 0.46674081683158875, 'Top': 0.5156008005142212}, 'Confidence': 94.0333282470703}], 'Parents': []}, {'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': 'ff25a927-93ab-40dd-9e8b-3e9d77d22062', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': 'ff25a927-93ab-40dd-9e8b-3e9d77d22062', 'content-type': 'application/x-amz-json-1.1', 'content-length': '2531', 'date': 'Mon, 01 May 2023 20:47:38 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 36 – Resultado da Figura 35 no terminal

- Saída do arquivo Excel:

Label (Rótulo)	Confidence	Detecção de rosto	Foto
City	99,88651	VERDADEIRO	praia.jpg
Person	98,86179	VERDADEIRO	praia.jpg
Metropolis	97,59953	VERDADEIRO	praia.jpg
Urban	97,59953	VERDADEIRO	praia.jpg
Clothing	97,34572	VERDADEIRO	praia.jpg
Hat	97,34572	VERDADEIRO	praia.jpg
Flag	94,03333	VERDADEIRO	praia.jpg
City	99,88651	VERDADEIRO	praia.jpg
Person	98,86179	VERDADEIRO	praia.jpg
Metropolis	97,59953	VERDADEIRO	praia.jpg

Figura 37 – Resultado da Figura 35 no Excel

6.3.2. Foto na neve: sem pessoas, até 10 camadas e o mínimo de confiança em 80%:

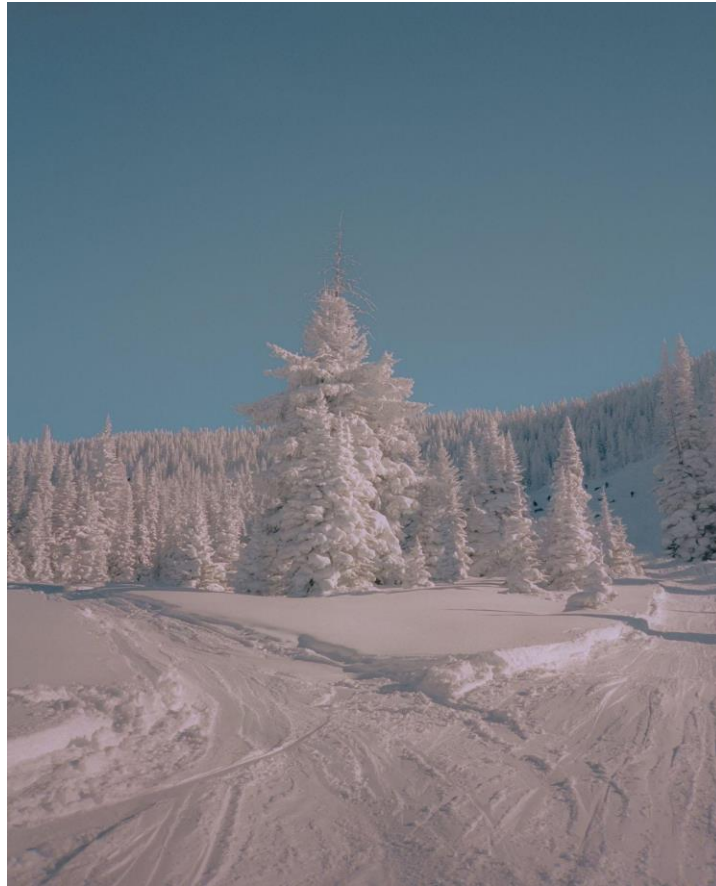


Figura 38 – Foto de paisagem 2

- Saída do Programa:

```
Não foram encontradas rostos na foto.
{'Labels': [{'Name': 'Plant', 'Confidence': 100.0, 'Instances': [], 'Parents': []}, {'Name': 'Tree', 'Confidence': 100.0, 'Instances': [], 'Parents': [{'Name': 'Plant'}]}, {'Name': 'Fir', 'Confidence': 99.99996185302734, 'Instances': [], 'Parents': [{'Name': 'Plant'}, {'Name': 'Tree'}]}, {'Name': 'Ice', 'Confidence': 99.36720275878906, 'Instances': [], 'Parents': []}, {'Name': 'Nature', 'Confidence': 94.34185791015625, 'Instances': [], 'Parents': [{'Name': 'Outdoors'}]}, {'Name': 'Outdoors', 'Confidence': 94.34185791015625, 'Instances': [], 'Parents': []}, {'Name': 'Weather', 'Confidence': 94.34185791015625, 'Instances': [], 'Parents': [{'Name': 'Nature'}, {'Name': 'Outdoors'}]}, {'Name': 'Pine', 'Confidence': 90.17996978759766, 'Instances': [], 'Parents': [{'Name': 'Plant'}, {'Name': 'Tree'}]}, {'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': '3ade77b4-581e-4910-b3c1-b242aeaa530a', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': '3ade77b4-581e-4910-b3c1-b242aeaa530a', 'content-type': 'application/x-amz-json-1.1', 'content-length': '1243', 'date': 'Mon, 01 May 2023 22:00:20 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 39 – Resultado da Figura 38 no terminal

- Saída do arquivo Excel:

Label (Rótulo)	Confidência	Deteccção	Foto
Plant	100	FALSO	neve.jpg
Tree	100	FALSO	neve.jpg
Fir	99,99996	FALSO	neve.jpg
Ice	99,3672	FALSO	neve.jpg
Nature	94,34186	FALSO	neve.jpg
Outdoors	94,34186	FALSO	neve.jpg
Weather	94,34186	FALSO	neve.jpg
Pine	90,17997	FALSO	neve.jpg

Figura 40 – Resultado da Figura 38 no Excel

- 6.3.3.** Miami no nascer do sol: até 20 camadas e o mínimo de confiança em 80%:



Figura 41 – Foto de paisagem 3

- Saída do programa:

```
Não foram encontradas rostos na foto.
{'Labels': [{'Name': 'City', 'Confidence': 99.99993896484375, 'Instances': [], 'Parents': []}, {'Name': 'Metropolis', 'Confidence': 99.99207305908203, 'Instances': [], 'Parents': []}, {'Name': 'Urban', 'Confidence': 99.99207305908203, 'Instances': [], 'Parents': []}, {'Name': 'Neighborhood', 'Confidence': 99.9913330078125, 'Instances': [], 'Parents': []}, {'Name': 'Palm Tree', 'Confidence': 99.9228515625, 'Instances': [], 'Parents': []}, {'Name': 'Plant', 'Confidence': 99.9228515625, 'Instances': [], 'Parents': []}, {'Name': 'Tree', 'Confidence': 99.9228515625, 'Instances': [], 'Parents': []}, {'Name': 'Road', 'Confidence': 99.90998840332031, 'Instances': [], 'Parents': []}, {'Name': 'Street', 'Confidence': 99.90998840332031, 'Instances': [], 'Parents': []}, {'Name': 'Architecture', 'Confidence': 99.57794189453125, 'Instances': [], 'Parents': []}, {'Name': 'Building', 'Confidence': 99.57794189453125, 'Instances': [], 'Parents': []}, {'Name': 'Condo', 'Confidence': 99.57794189453125, 'Instances': [], 'Parents': []}, {'Name': 'Housing', 'Confidence': 99.57794189453125, 'Instances': [], 'Parents': []}, {'Name': 'Path', 'Confidence': 99.54621124267578, 'Instances': [], 'Parents': []}, {'Name': 'Sidewalk', 'Confidence': 99.54621124267578, 'Instances': [], 'Parents': []}, {'Name': 'Cityscape', 'Confidence': 97.8945541381836, 'Instances': [], 'Parents': []}, {'Name': 'Summer', 'Confidence': 95.76024627685547, 'Instances': [], 'Parents': []}, {'Name': 'Person', 'Confidence': 91.05626678466797, 'Instances': [], 'Parents': []}, {'Name': 'High Rise', 'Confidence': 85.3902816772461, 'Instances': [], 'Parents': []}, {'Name': 'Fire Hydrant', 'Confidence': 83.32901000976562, 'Instances': [], 'Parents': []}], 'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': 'fa675ee8-550f-4071-b91c-54237204de63', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': 'fa675ee8-550f-4071-b91c-54237204de63', 'content-type': 'application/x-amz-json-1.1', 'content-length': '3972', 'date': 'Sun, 07 May 2023 15:31:34 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 42 – Resultado da Figura 41 no terminal

- Saída do Excel:

Label (Rótulo)	Confidence	Detecção de rosto	Foto
City	99,99994	FALSO	miami3.jpg
Metropolis	99,99207	FALSO	miami3.jpg
Urban	99,99207	FALSO	miami3.jpg
Neighborhood	99,99133	FALSO	miami3.jpg
Palm Tree	99,92285	FALSO	miami3.jpg
Plant	99,92285	FALSO	miami3.jpg
Tree	99,92285	FALSO	miami3.jpg
Road	99,90999	FALSO	miami3.jpg
Street	99,90999	FALSO	miami3.jpg
Architecture	99,57794	FALSO	miami3.jpg
Building	99,57794	FALSO	miami3.jpg
Condo	99,57794	FALSO	miami3.jpg
Housing	99,57794	FALSO	miami3.jpg
Path	99,54621	FALSO	miami3.jpg
Sidewalk	99,54621	FALSO	miami3.jpg
Cityscape	97,89455	FALSO	miami3.jpg
Summer	95,76025	FALSO	miami3.jpg
Person	91,05627	FALSO	miami3.jpg
High Rise	85,39028	FALSO	miami3.jpg
Fire Hydrant	83,32901	FALSO	miami3.jpg

Figura 43 – Resultado da Figura 41 no Excel

Uma curiosidade sobre a análise dessa foto: Ao olhar rapidamente para foto é quase certo de não ver uma pessoa nessa foto porque ela está bem atrás,

ainda assim o programa conseguiu identificar “person” como uma das camadas com um alto nível de confiança, de 91%.

6.3.4. Dia de carnaval, foto com resolução mais baixa, alguns rostos visíveis, até 10 camadas e o mínimo de confiança em 70%:



Figura 44 – Paisagem 4

- Saída do Programa:

```
Foram encontrados rostos na foto.
{'Labels': [{'Name': 'Carnival', 'Confidence': 99.97034454345703, 'Instances': [], 'Parents': []}, {'Name': 'Person', 'Confidence': 81.75269317626953, 'Instances': [{'BoundingBox': {'Width': 0.06260824203491211, 'Height': 0.1910010129213333, 'Left': 0.05868593230843544, 'Top': 0.8085111975669861}, 'Confidence': 81.75269317626953}, {'BoundingBox': {'Width': 0.07918623089790344, 'Height': 0.247090145945549, 'Left': 0.0, 'Top': 0.7528759241104126}, 'Confidence': 80.06056213378906}, {'BoundingBox': {'Width': 0.09570758044719696, 'Height': 0.16953691840171814, 'Left': 0.849316418170929, 'Top': 0.8300790190696716}, 'Confidence': 72.53577423095703}], 'Parents': []}, {'Name': 'Chandelier', 'Confidence': 80.77047729492188, 'Instances': [{'BoundingBox': {'Width': 0.9763651490211487, 'Height': 0.9789896011352539, 'Left': 0.018491361290216446, 'Top': 0.014175757765769958}, 'Confidence': 80.77047729492188}], 'Parents': [{'Name': 'Lamp'}]}, {'Name': 'Lamp', 'Confidence': 80.77047729492188, 'Instances': [], 'Parents': []}], 'LabelMode': 'Version': '3.0', 'ResponseMetadata': {'RequestId': 'b8a45d50-e9e3-4465-8db5-fd273ec76ead', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': 'b8a45d50-e9e3-4465-8db5-fd273ec76ead', 'content-type': 'application/x-amz-json-1.1', 'content-length': '1241', 'date': 'Mon, 01 May 2023 21:12:14 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 45 – Resultado da Figura 44 no terminal

- Saída do arquivo excel:

Label (Rótulo)	Confidenc	Detecção de rosto	Foto
Carnival	99,97034	VERDADEIRO	granderio2022.jpg
Person	81,75269	VERDADEIRO	granderio2022.jpg
Chandelier	80,77048	VERDADEIRO	granderio2022.jpg
Lamp	80,77048	VERDADEIRO	granderio2022.jpg

Figura 46 – Resultado da Figura 44 no Excel

6.4. Fotos de Desenho:

Foram utilizadas algumas imagens de desenhos para ver como o programa se comporta. Se ele lê esses desenhos como imagens/ fotos ou se ele realmente os vê como desenhos mesmos.

6.4.1. Desenho 1, 10 camadas e mínimo de confiança em 70%:



Figura 47 – Desenho 1

- Saída do Programa:

```
Não foram encontradas rostos na foto.
{'Labels': [{'Name': 'Stencil', 'Confidence': 99.31707763671875, 'Instances': [], 'Parents': []}, {'Name': 'Baby',
'Confidence': 98.15220642089844, 'Instances': [{'BoundingBox': {'Width': 0.5953575968742371, 'Height': 0.8419305682
182312, 'Left': 0.2016306072473526, 'Top': 0.07973373681306839}, 'Confidence': 98.15220642089844}], 'Parents': [{'N
ame': 'Person'}]}, {'Name': 'Person', 'Confidence': 98.15220642089844, 'Instances': [{'BoundingBox': {'Width': 0.59
53575968742371, 'Height': 0.8419305682182312, 'Left': 0.2016306072473526, 'Top': 0.07973373681306839}, 'Confidence'
: 98.15220642089844}], 'Parents': []}, {'Name': 'Face', 'Confidence': 95.46759796142578, 'Instances': [], 'Parents'
: [{'Name': 'Head'}, {'Name': 'Person'}]}, {'Name': 'Head', 'Confidence': 95.46759796142578, 'Instances': [], 'Pare
nts': [{'Name': 'Person'}]}, {'Name': 'Art', 'Confidence': 88.85565185546875, 'Instances': [], 'Parents': []}, {'Na
me': 'Drawing', 'Confidence': 88.85565185546875, 'Instances': [], 'Parents': [{'Name': 'Art'}]}, 'LabelModelVersio
n': '3.0', 'ResponseMetadata': {'RequestId': '66f2654e-a1f6-4c58-a735-4eb5135483ac', 'HTTPStatusCode': 200, 'HTTPHe
aders': {'x-amzn-requestid': '66f2654e-a1f6-4c58-a735-4eb5135483ac', 'content-type': 'application/x-amz-json-1.1',
'content-length': '1432', 'date': 'Mon, 01 May 2023 21:17:10 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 48 – Resultado da Figura 47 no terminal

- Saída do arquivo excel:

Label (Rótulo)	Confidência	Deteção	Foto
Stencil	99,31708	FALSO	desenho_pessoa2.jpg
Baby	98,15221	FALSO	desenho_pessoa2.jpg
Person	98,15221	FALSO	desenho_pessoa2.jpg
Face	95,4676	FALSO	desenho_pessoa2.jpg
Head	95,4676	FALSO	desenho_pessoa2.jpg
Art	88,85565	FALSO	desenho_pessoa2.jpg
Drawing	88,85565	FALSO	desenho_pessoa2.jpg

Figura 49 – Resultado da Figura 47 no Excel

6.4.2. Desenho 2: 10 camadas e mínimo de confiança em 70%:

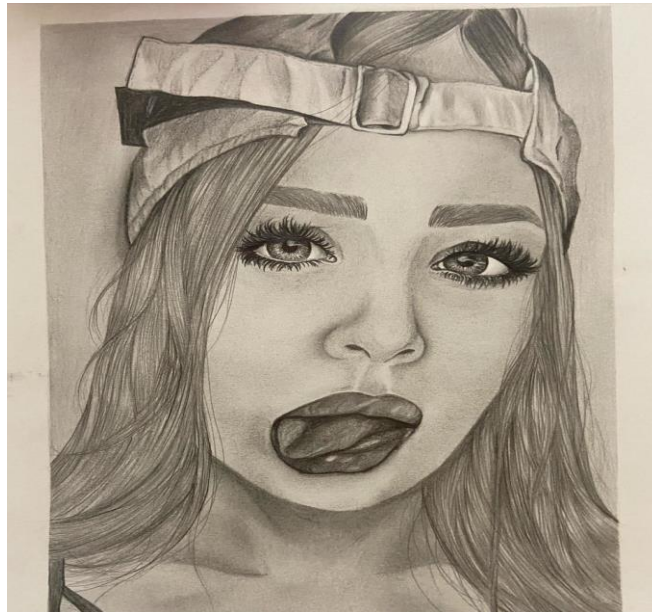


Figura 50 – Desenho 2

- Saída do Programa:

```
Foram encontrados rostos na foto.
{'Labels': [{'Name': 'Art', 'Confidence': 99.99974060058594, 'Instances': [], 'Parents': []}, {'Name': 'Drawing', 'Confidence': 99.99478149414062, 'Instances': [], 'Parents': [{'Name': 'Art'}]}, {'Name': 'Person', 'Confidence': 99.72230529785156, 'Instances': [{'BoundingBox': {'Width': 0.897067666053772, 'Height': 0.9891882538795471, 'Left': 0.05253833904862404, 'Top': 0.0071343486197292805}, 'Confidence': 99.72230529785156}], 'Parents': []}, {'Name': 'Face', 'Confidence': 98.10224151611328, 'Instances': [], 'Parents': [{'Name': 'Head'}, {'Name': 'Person'}]}, {'Name': 'Head', 'Confidence': 98.10224151611328, 'Instances': [], 'Parents': [{'Name': 'Person'}]}, 'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': '6324e51a-b470-4afb-b5c5-9b92d59841b0', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': '6324e51a-b470-4afb-b5c5-9b92d59841b0', 'content-type': 'application/x-amz-json-1.1', 'content-length': '990', 'date': 'Mon, 01 May 2023 21:35:52 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 51 – Resultado da Figura 50 no terminal

- Saída do arquivo excel:

Label (Rótulo)	Confidência	Detecção de rosto	Foto
Art	99,99974	VERDADEIRO	desenho_pessoa1.jpg
Drawing	99,99478	VERDADEIRO	desenho_pessoa1.jpg
Person	99,72231	VERDADEIRO	desenho_pessoa1.jpg
Face	98,10224	VERDADEIRO	desenho_pessoa1.jpg
Head	98,10224	VERDADEIRO	desenho_pessoa1.jpg

Figura 52 – Resultado da Figura 50 no Excel

Nos dois desenhos é possível ver um rosto, acontece que o segundo desenho é um desenho melhor elaborado que o primeiro desenho, e com isso o programa conseguiu detectar um rosto na função de detectar rostos, mas ainda assim o programa detectou “person” e “face” na primeira foto com bastante confiança, 98% e 95% respectivamente.

6.4.3. Desenho 3: 10 camadas e mínimo de confiança em 70%:

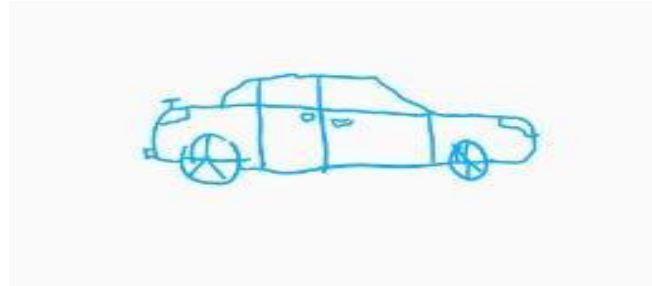


Figura 53 – Desenho 3

- Saída do Programa:

```
Não foram encontradas rostos na foto.
{'Labels': [{'Name': 'Art', 'Confidence': 96.99400329589844, 'Instances': [], 'Parents': []}, {'Name': 'Drawing', 'Confidence': 94.28573608398438, 'Instances': [], 'Parents': [{'Name': 'Art'}]}, 'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': '3915c95f-7841-486f-b1d7-223121b29d95', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': '3915c95f-7841-486f-b1d7-223121b29d95', 'content-type': 'application/x-amz-json-1.1', 'content-length': '366', 'date': 'Mon, 01 May 2023 21:43:16 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 54 – Resultado da Figura 53 no terminal

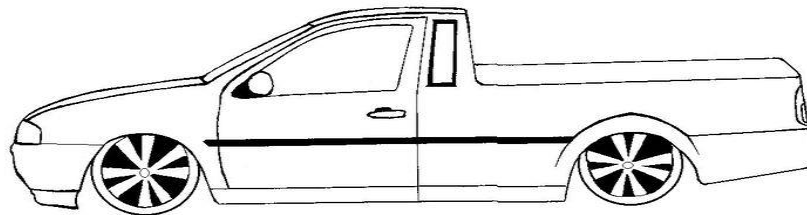
- Saída do arquivo excel:

Label (Rótulo)	Confidência	Deteção	Foto
Art	96,994	FALSO	desenho_carro2.jpeg
Drawing	94,28574	FALSO	desenho_carro2.jpeg

Figura 55 – Resultado da Figura 53 no Excel

Por ter sido um desenho mais simples, feito rapidamente na ferramenta Paint do computador, o programa só conseguiu reconhecer dois rótulos, mesmo colocando um nível de confiança de 70%.

6.4.4. Desenho 4:



www.desenhospainttool.com.br

Figura 56 – Desenho 4

• Saída do Programa:

```
Não foram encontradas rostos na foto.
{'Labels': [{'Name': 'Pickup Truck', 'Confidence': 99.99938201904297, 'Instances': [], 'Parents': [{'Name': 'Transportation'}, {'Name': 'Truck'}, {'Name': 'Vehicle'}]}, {'Name': 'Transportation', 'Confidence': 99.99938201904297, 'Instances': [], 'Parents': [{'Name': 'Truck', 'Confidence': 99.99938201904297, 'Instances': [], 'Parents': [{'Name': 'Transportation'}, {'Name': 'Vehicle'}]}, {'Name': 'Vehicle', 'Confidence': 99.99938201904297, 'Instances': [], 'Parents': [{'Name': 'Transportation'}]}, {'Name': 'Machine', 'Confidence': 91.23561096191406, 'Instances': [], 'Parents': [], 'Name': 'Wheel', 'Confidence': 91.23561096191406, 'Instances': [{'BoundingBox': {'Width': 0.12674908339977264, 'Height': 0.1732233613729477, 'Left': 0.6732245087623596, 'Top': 0.48851555585861206}, 'Confidence': 91.23561096191406}, {'BoundingBox': {'Width': 0.12791022658348083, 'Height': 0.1879509687423706, 'Left': 0.109576515853405, 'Top': 0.49465352296829224}, 'Confidence': 90.05481719970703}], 'Parents': [{'Name': 'Machine'}]}, {'Name': 'Car', 'Confidence': 79.73243713378906, 'Instances': [], 'Parents': [{'Name': 'Transportation'}, {'Name': 'Vehicle'}]}, {'Name': 'Limo', 'Confidence': 79.73243713378906, 'Instances': [{'BoundingBox': {'Width': 0.938736617565155, 'Height': 0.49849462509155273, 'Left': 0.015310644172132015, 'Top': 0.19317768514156342}, 'Confidence': 79.73243713378906}], 'Parents': [{'Name': 'Car'}, {'Name': 'Transportation'}, {'Name': 'Vehicle'}]}], 'LabelModelVersion': '3.0', 'ResponseMetadata': {'RequestId': 'eefbbcea-7f17-4f9e-89ea-9f696894b7dc', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': 'eefbbcea-7f17-4f9e-89ea-9f696894b7dc', 'content-type': 'application/x-amz-json-1.1', 'content-length': '1904', 'date': 'Mon, 01 May 2023 21:50:13 GMT'}, 'RetryAttempts': 0}}
Salvando o arquivo...
Arquivo salvo com sucesso!
```

Figura 57 – Resultado da Figura 56 no terminal

- Saída do arquivo excel:

Label (Rótulo)	Confidenc	Detecção	Foto
Pickup Truck	99,99938	FALSO	desenho_carro.jpg
Transportation	99,99938	FALSO	desenho_carro.jpg
Truck	99,99938	FALSO	desenho_carro.jpg
Vehicle	99,99938	FALSO	desenho_carro.jpg
Machine	91,23561	FALSO	desenho_carro.jpg
Wheel	91,23561	FALSO	desenho_carro.jpg
Car	79,73244	FALSO	desenho_carro.jpg
Limo	79,73244	FALSO	desenho_carro.jpg

Figura 58 – Resultado da Figura 56 no Excel

7. Capítulo 5: Considerações finais

Ao longo deste trabalho, foi possível explorar a aplicação da visão computacional utilizando os serviços da AWS, com foco no Amazon Rekognition. Foi evidente que a visão computacional tem desempenhado um papel fundamental no avanço da tecnologia, especialmente nos dispositivos móveis, onde a inteligência artificial e a visão computacional têm sido cada vez mais presentes. Os algoritmos de aprendizado de máquina otimizam o uso dos recursos e executam tarefas complexas em tempo real, proporcionando experiências enriquecedoras.

Dentre os serviços oferecidos pela AWS, o Amazon Rekognition destacou-se como uma ferramenta poderosa de visão computacional. Sua capacidade de analisar e reconhecer objetos, pessoas, texto e conteúdo inapropriado em imagens e vídeos demonstrou ser altamente eficiente. A identificação facial, análise de emoções, busca por conteúdo específico e outras funcionalidades proporcionaram resultados precisos e confiáveis nos testes feitos.

Diante do exposto, concluímos que a visão e fotografia computacional, em conjunto com os serviços da AWS, possui um enorme potencial para impulsionar a inovação e proporcionar melhores resultados nas aplicações fotográficas, de segurança onde esse serviço pode ajudar a identificar rostos,

como controle de acesso e monitoramento de multidões, reconhecimento de preços para compor relatórios, pesquisas e novos índices com base em mercados. A combinação da inteligência artificial, visão computacional e os recursos escaláveis da AWS abrem novas possibilidades para empresas e desenvolvedores que desejam criar soluções avançadas.

8. Referências:

HARRISON, Matt - **Marhine Learning Guia de Referência Rápida.**

2019, Disponível em:

<https://dokumen.pub/machine-learning-guia-de-referencia-rapida-trabalhando-com-dados-estruturados-em-python-9788575228173-857522817x.html>

BORGES, Luiz Eduardo - **Python Para Desenvolvedores.** 2014,

Disponível em:

https://books.google.com.br/books?hl=pt-BR&lr=lang_pt&id=eZmtBAAQBAJ&oi=fnd&pg=PA14&dq=o+que+%C3%A9+python&ots=VETopnIhjs&sig=OCHFJLbCFE9HOlgOFS6eA7yQwLE#v=onepage&q&f=false

TAULLI, Tom – **Introdução à Inteligência Artificial.** 2020

BARELLI, Felipe – **Introdução à Visão Computacional.** 2018, Disponível em:

<https://professor.luzerna.ifc.edu.br/ricardo-antonello/wp-content/uploads/sites/8/2017/02/Livro-Introdu%C3%A7%C3%A3o-a-Vis%C3%A3o-Computacional-com-Python-e-OpenCV-3.pdf>

SICHMAN, Jaime Simão - **Inteligência Artificial e sociedade: avanços e riscos.** 2021, Disponível em:

<https://www.scielo.br/j/ea/a/c4sqqrthGMS3ngdBhGWtKhh>

Documentação modulo CSV em Python. Disponível em:

<https://docs.python.org/3/library/csv.html>

Documentação biblioteca Openpyxl. Disponível em:

<https://openpyxl.readthedocs.io/en/stable/>

Documentação AWS Rekognition. Disponível em:

https://docs.aws.amazon.com/pt_br/rekognition/latest/dg/what-is.html

Documentação Boto3 AWS. Disponível em:

<https://boto3.amazonaws.com/v1/documentation/api/latest/index.html>