



DISSERTAÇÃO DE MESTRADO PROFISSIONAL

**Proposta de criação de banco de imagens de desaparecidos
e progressão de idade facial automática
usando Aprendizado de Máquina**

Aline dos Santos Pereira

Brasília, março de 2023

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

UNIVERSIDADE DE BRASÍLIA
Faculdade de Tecnologia

DISSERTAÇÃO DE MESTRADO PROFISSIONAL

**Proposta de criação de banco de imagens de desaparecidos
e progressão de idade facial automática
usando Aprendizado de Máquina**

Aline dos Santos Pereira

*Dissertação de Mestrado Profissional submetida ao Departamento de Engenharia
Elétrica como requisito parcial para obtenção
do grau de Mestre em Engenharia Elétrica*

Banca Examinadora

Prof. Daniel Chaves Café, Ph.D, FT/UnB
Orientador

Prof. Rafael Rabelo Nunes, Ph.D, FT/UnB
Examinador Interno

Maurício da Silva Sercheli, PCDF
Examinador externo

Prof. Demétrio Antônio da Silva Filho, IF/UnB
Examinador interno suplente

FICHA CATALOGRÁFICA

PEREIRA, ALINE DOS SANTOS

Proposta de criação de banco de imagens de desaparecidos e progressão de idade facial automática causando Aprendizado de Máquina [Distrito Federal] 2023.

xvi, 70 p., 210 x 297 mm (ENE/FT/UnB, Mestre, Engenharia Elétrica, 2023).

Dissertação de Mestrado Profissional - Universidade de Brasília, Faculdade de Tecnologia.

Departamento de Engenharia Elétrica

1. Aprendizado de Máquina

2. Segurança Pública

3. Progressão de idade

4. Redes Neurais

I. ENE/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

PEREIRA, A. D. S. (2023). *Proposta de criação de banco de imagens de desaparecidos e progressão de idade facial automática causando Aprendizado de Máquina*. Dissertação de Mestrado Profissional, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 70 p.

CESSÃO DE DIREITOS

AUTOR: Aline dos Santos Pereira

TÍTULO: Proposta de criação de banco de imagens de desaparecidos e progressão de idade facial automática causando Aprendizado de Máquina.

GRAU: Mestre em Engenharia Elétrica ANO: 2023

É concedida à Universidade de Brasília permissão para reproduzir cópias desta Dissertação de Mestrado Profissional e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. Os autores reservam outros direitos de publicação e nenhuma parte dessa Dissertação de Mestrado Profissional pode ser reproduzida sem autorização por escrito dos autores.

Aline dos Santos Pereira

Depto. de Engenharia Elétrica (ENE) - FT

Universidade de Brasília (UnB)

Campus Darcy Ribeiro

CEP 70919-970 - Brasília - DF - Brasil

AGRADECIMENTOS

A todos que me deram uma chance, me ensinaram e me ajudaram no desenvolvimento deste trabalho. Sem vocês, esse feito não seria alcançado.

RESUMO

Nos últimos 5 anos, no Brasil, em média 200 pessoas desaparecem por dia. Quando uma pessoa desaparece, as autoridades policiais realizam um levantamento a respeito da aparência da pessoa em questão e resulta em uma fotografia para a divulgação de desaparecimento ou em um retrato falado. Com o passar do tempo, esses registros fotográficos se tornam desatualizados por conta do envelhecimento natural, principalmente em caso de crianças desaparecidas. Existem diversas técnicas de progressão de idade para acompanhar essas mudanças. No Brasil, a legislação obriga a atualização de imagens de progressão de idade para todos os desaparecidos. Para aqueles com menos de 18 anos com uma frequência a cada três anos e após 18 anos, a cada cinco anos. Porém, existem vários fatores que dificultam esse processo de atualização dessas imagens, como volume de imagens e ausência de um banco de dados unificado. Para estes problemas, automação utilizando aprendizado de máquina pode ser uma alternativa adicional, mais especificamente, utilização das Redes Adversárias Generativas (GANs). Para isso, é necessário um banco de dados padronizado dos cidadãos brasileiros nesta situação de desaparecimento. Apesar de existir uma lei que institui a unificação dessas informações de desaparecidos, incluindo imagens, o Governo Brasileiro e os órgãos envolvidos neste processo ainda estão longe de atingir o resultado final do que a lei impõe. Este trabalho demonstra como forças de segurança pública Brasileiras lidam com dados de pessoas desaparecidas, e realça a importância e a urgência da existência de um banco de dados unificado. Adicionalmente, demonstra como a utilização de novas técnicas de manipulação de imagem, incluindo as Redes Neurais, podem auxiliar forças policiais durante a investigação de crianças e jovens desaparecidos. Foi possível demonstrar como a informação é disseminada pelo país de acordo com cada estado. Adicionalmente, obter resultados promissores em questão de acurácia entre imagens manipuladas por redes neurais e criar um banco de imagens de vítimas brasileiras.

ABSTRACT

Within the last five year, in Brazil, an average of 200 people disappeared per day. When someone disappears, the policial authorities make a research about the person's physical appearance. This research may result in a photography for divulgation purposes about the persons's missing or a police sketch. Through the pass of time, theses portraits get outdated due to aging, specially in cases of missing children. There is a multitude of techniques for age progressing of all of the missing people. Brazilian's legislation demands an update of the images of age progressing for all of the missing people. For those who are less than 18 years-old the update must be made every three year, and after 18 years-old every five years. However, there are a lot of factors that make this procedure difficult such as the volume of images and the absence of a database. These problems may be approached by using automation through machine's learning, more specifically using Generative Adversarial Networks (GANs). For this purpose, a data bank of missing Brazilian citizens would be necessary. Even though there is Brazilian legislation instituting the unification of

missing people's information, including images, the Brazilian government and the authorities involved in this procedure are yet far from getting the final results imposed by law. This paper demonstrates how Brazilian authorities deal with missing people's data and emphasizes the importance and urgency of a unified database. Furthermore, this paper demonstrates how the usage of new techniques of image's manipulation may aid policial authorities during a missing investigation of children and teenagers. It was possible to demonstrate how the disappearances information is disseminated across the country. In addition, it was possible to obtain promising results in terms of accuracy between images generated by neural networks and to create an image database of Brazilian victims.

SUMÁRIO

1	INTRODUÇÃO	1
1.1	MOTIVAÇÃO	2
1.2	OBJETIVOS	2
1.3	CONTRIBUIÇÕES	3
1.4	ORGANIZAÇÃO	3
2	REVISÃO BIBLIOGRÁFICA	4
2.1	<i>Deep Learning</i>	4
2.2	REPRESENTAÇÃO HUMANA FACIAL	14
2.3	BANCOS DE IMAGENS FACIAIS	15
3	TRABALHOS RELACIONADOS	16
3.1	REDES ADVERSÁRIAS GENERATIVAS	16
3.2	RECONHECIMENTO FACIAL HUMANO	18
4	METODOLOGIA	20
4.1	ACESSO À INFORMAÇÃO	20
4.2	EXTRAÇÃO MANUAL E AUTOMÁTICA DAS IMAGENS	24
4.3	ARQUITETURA PROPOSTA	26
4.4	AMBIENTE DE SIMULAÇÃO	27
5	RESULTADOS	35
5.1	DISPONIBILIZAÇÃO DOS DADOS E QUALIDADE DE INFORMAÇÕES	35
5.2	BANCO DE DADOS	38
5.3	COMPARAÇÃO COM CASOS REAIS	39
6	CONCLUSÃO	45
	REFERÊNCIAS BIBLIOGRÁFICAS	47
	APÊNDICES	52
I.1	TABELAS DE CLASSIFICAÇÃO POR IMAGEM POR CADA ESTADO BRASILEIRO EXTRAÍDO MANUALMENTE	52
I.2	CÓDIGO PYTHON DE EXTRAÇÃO DE IMAGENS	60

LISTA DE FIGURAS

2.1	Operações lógicas feitas por neurônios artificiais	5
2.2	Arquitetura MLP.....	6
2.3	Arquitetura CNN.	7
2.4	Exemplo de uma tradução de imagem feita por um VAE.....	8
2.5	Evolução de qualidade em relação às imagens geradas por GANs.....	9
2.6	Exemplo do site uma imagem de um rosto gerado por uma GAN do site "Essa pessoa não existe".	10
2.7	Arquitetura de funcionamento de uma GAN.....	11
2.8	Exemplo de tradução de imagens feitas por GANs.....	12
2.9	Tradução feitas pela <i>StarGAN</i> , iniciando com quatro entradas, é possível traduzir em diversos domínios sejam eles de características fenotípicas, sejam expressões faciais.	13
2.10	Tradução feitas pela <i>StyleGAN</i> de imagens de entrada (<i>Source A</i>) em um estilo das outras imagens <i>Source B</i>	14
3.1	Comparação entre outros modelos de tradução de imagem e a <i>AttentionGAN</i> proposta por H. Tang et al.	17
4.1	Sinesp Cidadão: Primeira tela do aplicativo onde é possível filtrar as informações das pessoas desaparecidas.	21
4.2	Sinesp Cidadão: Lista de pessoas disponível no aplicativo e exemplo do detalhamento de uma delas.....	22
4.3	Comparação entre uma imagem disponibilizada pela Polícia Civil do Amazonas (Amazonas) e uma disponibilizada pelo SINALID Alagoas (PLID/AL).	25
4.4	Exemplo do processo utilizado pelas Polícias Cíveis. Imagens reais retiradas da Polícia Civil do Distrito Federal.....	26
4.5	Arquitetura proposta utilizando um banco de dados unificado e algoritmos de aprendizado de máquina.	27
4.6	Comparação entre os resultados de tradução de imagem em diversos domínios feitos pela <i>AttentionGAN</i> e <i>StarGAN</i>	28
4.7	Resultados prévios obtidos durante a metade do treino da <i>AttentionGAN</i> seguindo os passos disponíveis no repositório do trabalho de L. Hao et al.	29
4.8	Início dos treinos da <i>AttentionGAN</i> com as imagens da Polícia Civil do Distrito Federal.	30
4.9	Resultados de envelhecimento obtidos por A. Chandra em seu artigo.....	31
4.10	Resultados reproduzidos do projeto de A. Chandra (Matcha 2019) utilizando o UTKFace....	32
4.11	Resultados reproduzidos do projeto de A. Chandra (Matcha 2019) utilizando o banco de dados Brasileiro manualmente construído.....	32
4.12	Menus e modificações disponíveis no aplicativo <i>FaceApp</i>	33
4.13	Ferramenta de Comparação Facial disponível em <i>Face++</i> (<i>Face++</i> 2017).....	34
5.1	Contagem imagens por estados	37

5.2	Comparação entre as progressões de imagem da Polícia e imagens do <i>FaceApp</i>	40
5.3	Criança 1: Comparação entre a progressão de imagem da Polícia Civil do Distrito Federal e o aplicativo <i>FaceApp</i>	41
5.4	Criança 2: Comparação entre a progressão de imagem da Polícia Civil do Distrito Federal e o aplicativo <i>FaceApp</i>	42
5.5	Criança 3: Comparação entre a progressão de imagem original da criança e do aplicativo <i>FaceApp</i> . Logo, a comparação da imagem original da criança com 11 anos e novamente com a atual com 21 anos.....	43

LISTA DE TABELAS

4.1	Fontes de dados de cada estado Brasileiro	23
5.1	Análise das informações disponíveis sobre pessoas desaparecidas de cada estado brasileiro .	36
5.2	Análise das informações disponíveis sobre pessoas desaparecidas de cada estado brasileiro .	39
1	Classificação das imagens do estado da Bahia	52
2	Classificação das imagens do estado do Distrito Federal	55
3	Classificação das imagens do estado do Rio de Janeiro	56
4	Classificação das imagens do estado do Paraná	58

LISTA DE SÍMBOLOS

Siglas

DNA	<i>Ácido Desoxirribonucleico</i>
GAN	<i>Generative Adversarial Network</i>
RNA	<i>Redes Neurais Artificiais</i>
SVM	<i>Support Vector Machines</i>
TLU	<i>Threshold Logic Unit</i>
MLP	<i>Multi-Layer Perceptron</i>
DNN	<i>Deep Neural Networks</i>
CNN	<i>Convolutional Neural Networks</i>
VAE	<i>Variational Autoencoder</i>
PCA	<i>Principal Component Analysis</i>
LDA	<i>Linear Discriminant Analysis</i>
RIBPG	<i>Rede Integrada de Bancos de Perfis Genéticos</i>
FRT	<i>Facial Recognition Technology</i>
SINESP	<i>Sistema Nacional de Informação de Segurança Pública</i>

1 INTRODUÇÃO

De acordo com L. Roy and S. Arunachalam (ROY e ARUNACHALAM), o crescimento de empresas *startups* que oferecem serviço de análise de dados, geração de dados e vigilância está acompanhando a necessidade de instituições de segurança pública. Utilizando algoritmos de inteligência artificial, essas empresas conseguem prevenir e auxiliar o cometimento e resolução de crimes. Existem até mesmo aplicações úteis quando o assunto é preservação de identidade e identificação de autoria. Durante a pandemia do COVID-19, a tecnologia ajudou na identificação de cidadãos que quebravam protocolos de quarentena (ROY e ARUNACHALAM).

Quando se trata de identificação de pessoas desaparecidas, atualmente, existem diversas técnicas, porém, o tema Aprendizado de Máquina ainda não é amplamente disseminado. Especialmente dentro das forças Brasileiras policiais e de segurança. De acordo com o catálogo de Segurança Pública do Brasil (Pública 2022), mais de 65 mil ocorrências de desaparecimento foram registrados apenas no ano de 2021. Ainda no Brasil, ao longo dos últimos cinco anos, são ao menos 200 casos por dia. C. Acayaba discorre sobre a obrigatoriedade, imposta pela legislação Brasileira, da condução de investigações para crianças e adolescentes. Contudo, na prática, esse compromisso com a lei ocorre apenas para casos onde a vítima possui até 12 anos de idade (Acayaba). Tendo em vista que muitos casos não são investigados somados à maturação natural da aparência de jovens e crianças, certamente, os dados e imagens dessas pessoas ficam desatualizados.

Em alguns estados brasileiros, está disponível diversos dados a respeito de pessoas desaparecidas, incluindo as imagens de progressão de idade previstas por lei. Para crianças e adolescentes com até 18 anos, as técnicas de progressão devem ser aplicadas a cada três anos e os demais, a cada cinco anos (Comunicação/DGPC). O profissional responsável por aplicar essas técnicas é o de identificação humana, ou seja, um papiloscopista. Quando a progressão de idade se faz necessária, são utilizados três processos principais. Primeiramente, é feito um estudo acerca da vida da vítima, como hábitos únicos e familiares, enfermidades que podem acometer aparência. Em sequência, são reunidas diversas fotos onde é possível observar o rosto da vítima em um ângulo de frente. Por último, é feita uma manipulação em *Photoshop* da foto escolhida de acordo com o ângulo e qualidade. Nesta manipulação, o papiloscopista deve ser fiel ao estudo do estilo de vida da vítima, da família e a fisiologia natural da pessoa (Comunicação/DGPC).

Mesmo com o esforço das polícias, as técnicas citadas podem ser aprimoradas quando combinadas com o Aprendizado de Máquina. Existem casos onde redes neurais foram utilizadas para identificação de pessoas. Um bom exemplo é o caso chinês onde um homem foi identificado após anos desaparecido (Gan). De acordo com N. Gan, Mao Yin foi sequestrado ainda muito pequeno e graças a uma rede neural de reconhecimento ele foi, primeiramente, identificado. Após essa identificação, um teste de DNA conseguiu reunir Mao Yin à sua família biológica (Gan). Casos como o de Mao Yin podem ser reproduzidos caso técnicas de aprendizado de máquina sejam adicionadas ao procedimento usual da polícia brasileira.

Embora existam muitas abordagens de Redes Neurais que consigam auxiliar nessa questão de progressão de idade, é necessário um banco de dados condizente com o fenótipo da população brasileira

para testar e aplicar tais técnicas. Cada um dos 26 estados brasileiros e o Distrito Federal possui sua própria diligência de investigação em casos de pessoas desaparecidas. Apenas em 2019 foi sancionada a Lei 13.812/2019 (República), onde institui a Política Nacional de Busca de Pessoas Desaparecidas e uma cooperação de órgãos investigativos da federação e do Distrito Federal através de um cadastro único nacional de pessoas desaparecidas. Dentro do texto da lei, é também instituído o apoio à pesquisa e novas tecnologias que possam avançar no cumprimento da lei e na criação do resultado final, o banco de dados.

Exposta as diversidades e possibilidades dentro do cenário brasileiro de pessoas desaparecidas, o objetivo deste trabalho é propor uma arquitetura onde auxilie técnicas existentes de envelhecimento de rostos em imagens. Nesta arquitetura, foi necessário trabalhar com diversos bancos de dados, criar um banco de dados de imagens brasileiras com as informações disponíveis e testá-los com as redes neurais e outras ferramentas.

1.1 MOTIVAÇÃO

Considerando que o banco de dados nacional ainda não existe, este trabalho usa esse fato como motivação para demonstrar a importância da existência desse banco dentro das conformidades da Lei. Ademais, considerando que as técnicas de redes neurais avançam cada dia mais e existe uma escassez do quadro policial (Pública 2022), essas técnicas podem ser ferramentas aliadas para estados que não possuem nenhum tipo de diligência para imagens de progressão de idade ou mesmo divulgação de informações sobre seus desaparecidos e estatísticas. Este segundo fato é uma motivação para estudar e comparar as técnicas da literatura, mostrando que com as informações reunidas, técnicas de *Deep Learning* e uma nova arquitetura do processo de progressão de idade a polícias poderiam se beneficiar.

1.2 OBJETIVOS

É possível esboçar uma solução utilizando redes neurais que some às técnicas de progressão de idade aplicadas no Brasil. A utilização de uma GAN, por exemplo, pode ser uma ferramenta rápida e escalável para auxiliar a identificação de pessoas desaparecidas. Por ser um processamento rápido, permite atualizar tais imagens em um intervalo de tempo menor. O primeiro objetivo deste trabalho é reunir as fontes e criar um banco de dados. Dessa maneira, é possível demonstrar a importância de haver um banco de dados unificado e com boas imagens que seja possível de treinar e testar as Redes Neurais. O segundo objetivo é demonstrar técnicas de progressão automática da idade facial usando GANs como o *AttentionGAN* (Tang et al. 2021), *CycleGAN* (Matcha 2019) e (Goncharov 2017) somado ao Face++ (Face++ 2017).

1.3 CONTRIBUIÇÕES

As contribuições deste trabalho são:

1. Mapear e avaliar as fontes brasileiras sobre a divulgação de informações de pessoas desaparecidas;
2. Criar um banco de imagens de vítimas Brasileiras;
3. Aplicar diferentes técnicas utilizando GANs com esse banco de imagens;
4. Demonstrar o quão benéfico seria na criação de um banco de dados unificado e ter uma interoperabilidade de técnicas para auxiliar na identificação dessas pessoas desaparecidas.

1.4 ORGANIZAÇÃO

Este trabalho está dividido por capítulos, onde o segundo capítulo apresenta uma relevante revisão acerca da área de estudo abordado, como Aprendizado Profundo, Redes Neurais Adversárias (GANs) e Reconhecimento humano facial. O terceiro capítulo discorre a respeito de trabalhos relacionados a proposta deste trabalho. O quarto apresenta toda a metodologia aplicada em virtude de obter os resultados devidos, descritos no capítulo cinco. Por último, a conclusão do trabalho e propostas futuras. Existem Apêndices ao final do trabalho onde contém tabelas e códigos que auxiliam no entendimento deste trabalho e trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA

Para compreender este trabalho, é importante entender conceitos relacionados à área de Aprendizado de Máquina. Neste capítulo, é apresentada uma breve revisão do que é *Deep Learning* e alguns tipos de Redes Neurais. Além disso, é citado também algumas bibliotecas do *Python* e ferramentas relevantes disponíveis para o público em geral. Bancos de dados amplamente disseminados e alimentados. Por fim, são apresentados conceitos e técnicas de reconhecimento facial humano.

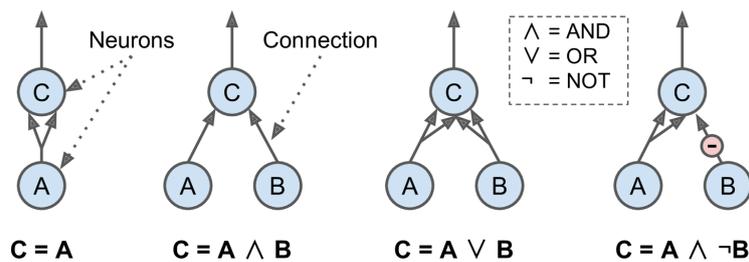
2.1 DEEP LEARNING

Resumidamente, *Deep Learning* é uma subárea do Aprendizado de máquina onde tarefas muito complexas são executadas. *Deep Learning* utiliza parâmetros básicos em um conjunto de dados para treinar um computador a aprender por conta própria, reconhecendo padrões em várias camadas de processamento~(Géron 2017). Esta visão do aprendizado de máquina ampliou a sua utilização. Hoje em dia é possível reconhecer padrões com mais precisão e reproduzir, por exemplo, pinturas de estilos famosos ou sinfonias em que os humanos não conseguem fazer com a mesma facilidade. As Redes Neurais Artificiais (RNA) são o núcleo principal do *Deep Learning*. Diferente do que muitos pensam, essas redes foram introduzidas em 1943 por Warren McCulloch e Walter Pitts (Géron 2017). Ambos argumentaram que era possível realizar cálculos complexos utilizando lógica proposicional. Desde então, nas décadas de 1980 e 1990 muitas outras arquiteturas foram inventadas. Diferente do passado, atualmente existe poder computacional suficiente para desenvolver e treinar RNAs que consigam enfrentar problemas muito complexos e com uma grande quantidade de dados.

Redes Neurais Artificiais

Como citado anteriormente, as RNAs vêm a partir de um conceito muito antigo e portanto, muitas arquiteturas foram criadas desde então. A primeira, McCulloch e Pitts propuseram o que hoje é conhecido como neurônio artificial (Géron 2017). É um modelo muito simples baseado em neurônios biológicos. Em computação, os neurônios são ativados quando há uma determinada entrada, seja um valor 0 ou 1. Utilizando este modelo, é possível operar muitos outros cálculos lógicos como AND, OR, NOT (Géron 2017).

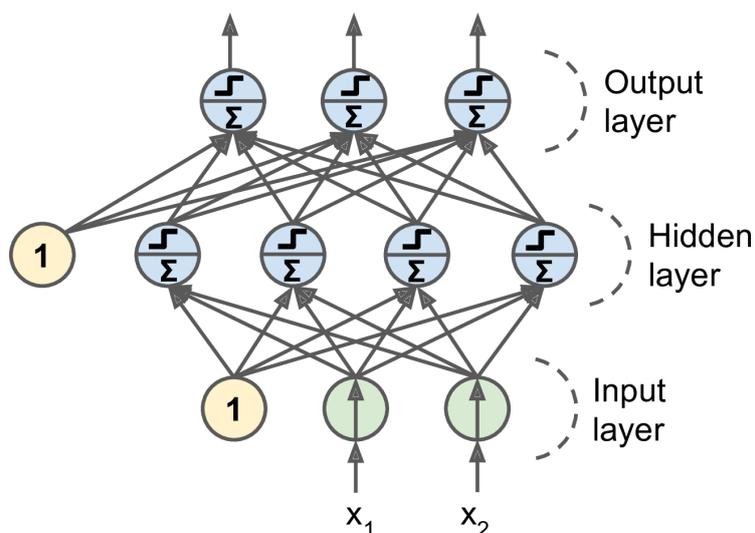
Figura 2.1: Operações lógicas feitas por neurônios artificiais



Fonte: (Géron 2017).

Na Figura 2.1, para o segundo esquema, os neurônios são ativados quando os anteriores são ativados. Em outras palavras, para esta operação AND, o neurônio C é ativado apenas se os neurônios A e B estiverem ativados com o valor 1. Outro exemplo, para a operação identidade na imagem 2.1, o primeiro esquema da esquerda para a direita, o neurônio C só será ativado caso A seja ativado, valor 1. Mais tarde, em 1957, Frank Rosenblatt propôs a arquitetura de perceptron (Géron 2017). Deslocando a lógica liga/desliga dos neurônios artificiais, as entradas agora são qualquer número e a nova arquitetura calcula uma soma de suas entradas com pesos apropriados, então é aplicada uma função degrau. Essa arquitetura é chamada de *Threshold Logic Unit* (TLU), onde é possível obter saídas de classificação onde há apenas uma camada de perceptrons e todos os neurônios estão conectados a todas as entradas. Mais tarde, em 1986, Rumelhart et. al publicou um artigo abordando como treinar o *Multi-Layer Perceptron* (MLP) (Rumelhart, Hinton e Williams 1986). A arquitetura MLP é composta por uma camada de entrada, pelas camadas ocultas e por fim, a camada de saída. Uma vez que uma MLP é uma Rede Neural Profunda por conter duas ou mais camadas ocultas, porém, naquela época era muito difícil treinar uma MLP pois não havia poder computacional suficiente. A Figura 2.2 representa uma MLP onde é composta por uma camada de entrada, X1 e X2, a próxima é a camada oculta e por último, a camada de saída. Quando uma MLP possui mais de uma camada oculta, já pode ser considerada uma Rede Neural Profunda.

Figura 2.2: Arquitetura MLP.



Fonte: (Géron 2017)

Rumerlhart et. al propuseram como treinar MLPs com os conceitos de *backpropagation* que consiste em cada neurônio calcular a saída e passar para o próximo neurônio da próxima camada e assim por diante. O erro total de saída da rede é calculado a cada neurônio para obter o quão distante toda a rede está da saída desejada. Este cálculo é feito até que a camada de entrada seja atingida. Esta técnica reversa permite que as informações sejam propagadas por toda a rede para efetivar o recálculo dos pesos de cada neurônio (Géron 2017).

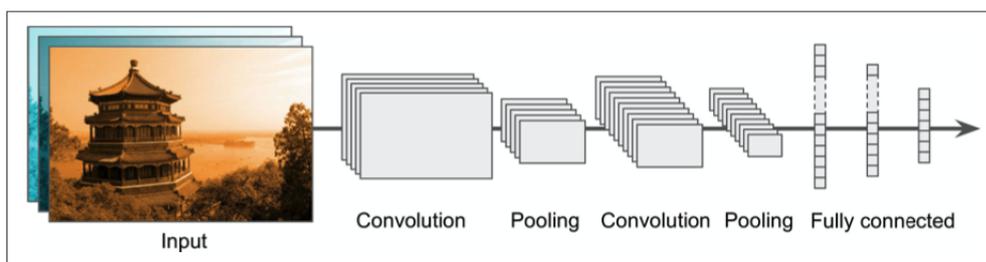
Atualmente, mesmo com diversas arquiteturas, as Redes Neurais Profundas ou em inglês, *Deep Neural Networks*(DNNs) são baseadas em MLPs. Cada DNN com sua próprias particularidades. Tais conceitos foram primordiais para o desenvolvimento das redes neurais de hoje, principalmente em questão de atingir resultados que o olho humano não consiga distinguir. Neste trabalho, o foco principal são as Redes Adversárias Generativas das quais descendem de MLPs mas ainda estão mais próximas das Redes Neurais Convolucionais (CNNs) e os *Autoencoders* Variacionais (VAEs).

A origem das CNNs advém de tentativas em imitar a visão humana, o que seria tarefa muito difícil. De acordo com M. Sewak et al. (Sewak, Karim e Pujari 2018), o primeiro algoritmo de reconhecimento visual foi feito nos anos 70 em um laboratório de Inteligência Artificial na Universidade de *Stanford*, mas o primeiro algoritmo de reconhecimento foi efetivamente utilizado em 2006 em uma câmera *Fuji-Film* (Sewak, Karim e Pujari 2018). Basicamente, as CNNs são um tipo especial de rede multicamadas. O que a difere de outras redes é o seu *design*, com mais camadas ocultas, resultando em uma performance melhor. As CNNs foram feitas pela primeira vez para reconhecer padrões com processamento mínimo. Como nas Redes Neurais comuns, uma CNN recebe um vetor como entrada e passa por todas as camadas ocultas e seus neurônios com seus respectivos pesos. Na Figura 2.3, é possível observar uma arquitetura comum de uma CNN. Geralmente são três camadas principais: camada de *convolução*, camada de *pooling* e camada *fully connected* (Sewak, Karim e Pujari 2018). A primeira camada de *convolução* não está

conectada em cada pixel de uma imagem. Apenas na parte principal de uma imagem, pro exemplo, um rosto. Por sua vez, os seguintes neurônios da segunda camada convolucional são conectados apenas aos neurônios localizados dentro de um pequeno retângulo na primeira camada, nesse exemplo, um retângulo em cima do rosto a ser reconhecido. Essa arquitetura permite que a rede se concentre em pequenos recursos de baixo nível na primeira camada oculta e, em seguida, reúna-os em recursos maiores de nível superior na próxima camada oculta e assim por diante (Géron 2017). Resumidamente, as camadas convolucionais conseguem delimitar as partes importantes de uma imagem, quanto mais camadas convolucionais, mais detalhada é essa identificação e o mapeamento dos traços importantes da imagem.

Após as camadas convolucionais, existe a etapa de *pooling*. Essa camada simplifica a informação da camada anterior, como as camadas convolucionais transformam vetores de características em matrizes de mapeamento. Essa camada *Pooling* resume toda a informação de uma área da imagem mapeada em forma de matriz em um único valor para uma nova matriz e assim por diante. Após todos esses processos, como pode ser visto na Figura 2.3, existe a última etapa chamada *fully connected* onde N entradas são o resultado final das camadas convolucionais adicionais as de *pooling*, sendo a sua saída N classificações.

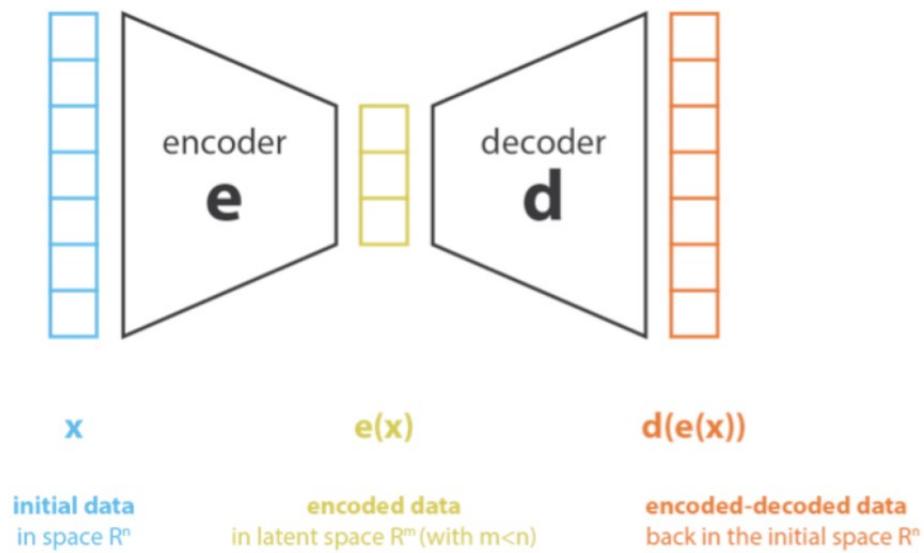
Figura 2.3: Arquitetura CNN.



Fonte: (Géron 2017)

Autoencoders Variacionais, ou simplesmente VAEs, são modelos generativos, mas funcionam de forma diferente das GANs. O objetivo do VAE é produzir resultados utilizando o que é chamado de espaço latente. Ao contrário das GANs, o VAE tenta modelar a entrada em vez de encontrar um modelo que se aproxime da mesma (Atienza 2018). Ou seja, os VAEs geram novos dados através de sua codificação ajustadas por treinamento, dessa forma, é possível ter um espaço latente que performam melhor para gerar esses dados. O funcionamento dos VAEs é baseado em um codificador e um decodificador. Primeiro, utilizando este método, há um processo para codificar características especiais de dados e é utilizada uma propriedade chamada redução de dimensionalidade de Aprendizado de Máquina. O objetivo dessa redução é encontrar um par perfeito de codificação/decodificação. Este par deve ser fácil de codificar e não difícil o suficiente para decodificar, mas mantendo o máximo possível de informações. A responsabilidade do decodificador é percorrer os dados codificados no espaço latente e decodificar esses dados fornecendo as saídas desejadas. O espaço latente é o mundo de possibilidades, dentro desse espaço existem diversos pontos, vetores, com tais características que determinam, no nosso exemplo, uma face. Cada face encontrada pelo *encoder* é um ponto no espaço latente. Essa arquitetura de entrada, codificador, seleção de um vetor no espaço latente e decodificação está ilustrada na Figura 2.4.

Figura 2.4: Exemplo de uma tradução de imagem feita por um VAE.



Fonte: (Atienza 2018)

Redes Adversárias Generativas

Apesar de muito recente, as arquiteturas GANs já possuem muitas aplicabilidades. Porém, foram primeiramente introduzidas em 2014 por I. Goodfellow et al. em forma de artigo (Goodfellow et al. 2014). E em 2019, I. Goodfellow fez uma postagem em sua rede social, *Twitter*, onde demonstrava a evolução das GANs ao longo desses quase cinco anos.

Figura 2.5: Evolução de qualidade em relação às imagens geradas por GANs.

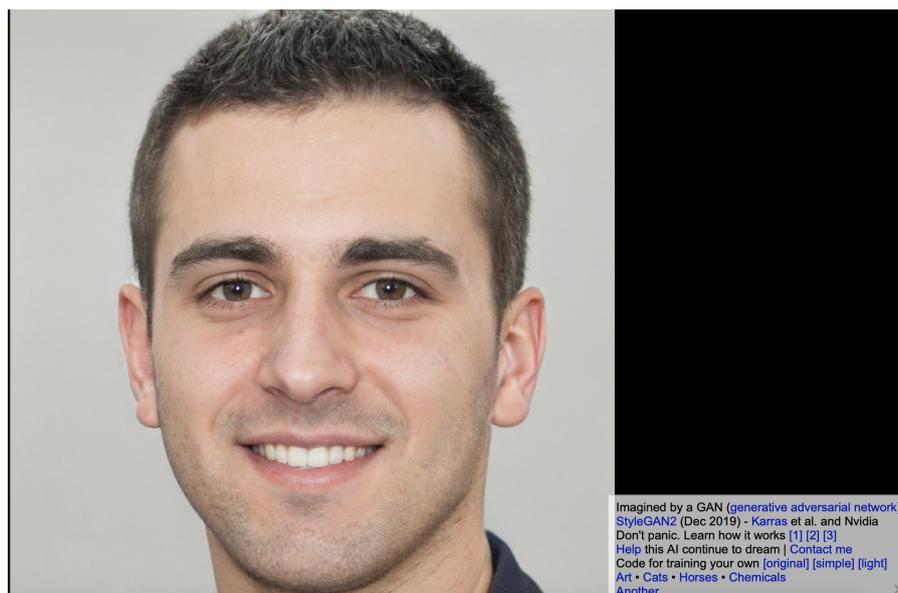


Fonte: (Goodfellow)

Como observado pelo *tweet* de Ian Goodfellow, atualmente, as GANs são capazes de gerar imagens, pinturas, músicas e muitas outras coisas extremamente realistas. Diferentemente dos modelos anteriores, cujo objetivo é classificar entre as classes, as GANs utilizam um modelo Generativo, onde o objetivo principal é tentar aprender e representar uma classe o mais realista possível, criando novos dados.

Este tipo de rede pode gerar imagens que olhos humanos não conseguem diferenciar. Um ótimo exemplo é o site "Esta pessoa não existe" (Karras et al. 2019).

Figura 2.6: Exemplo do site uma imagem de um rosto gerado por uma GAN do site "Essa pessoa não existe".



Fonte: (Karras et al. 2019)

Como as Redes Adversárias Generativas funcionam?

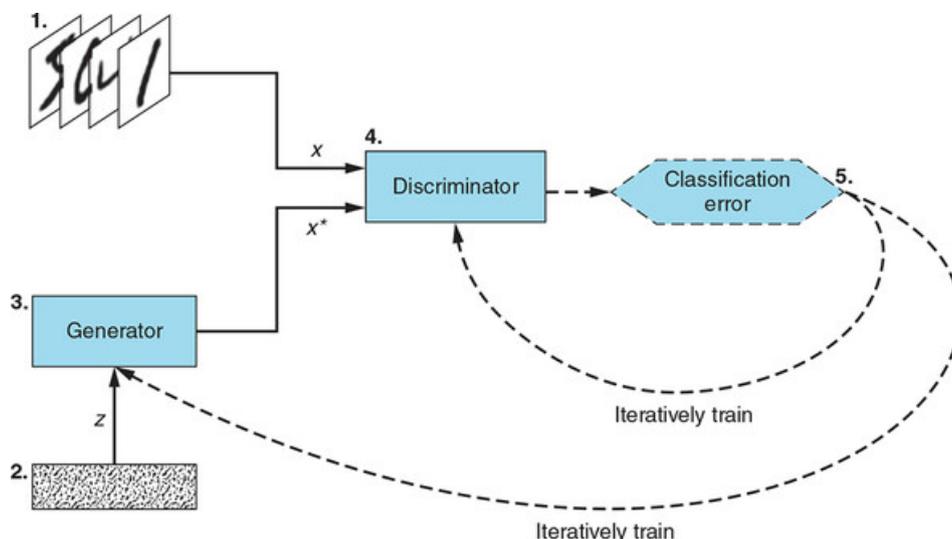
O funcionamento de uma GAN é baseado em uma competição entre dois modelos. Ou seja, as GANs são compostas por dois modelos treinados diferentes, um é o gerador e o outro é o discriminador. Os dados resultantes que uma GAN pode gerar dependem de um conjunto de dados escolhido; portanto, o gerador recebe uma entrada adicionada a um ruído e então consegue gerar imagens tão reais quanto possível naquele ponto do treinamento. O vetor de ruído como entrada é muito importante pois é um vetor aleatório de *features*, ou seja, a cada nova entrada, esse vetor é composto aleatoriamente de zeros ou uns. Com os valores desse vetor, a GAN sabe as características do que ela deve gerar a partir desse vetor de ruído. Por exemplo, um vetor possui um tamanho de 128, e dentro desses 128 valores, o valor 1 em uma posição X determina cabelos longos. Logo, a rede vai gerar uma nova imagem de uma pessoa com cabelos longos. É por isso que para todo início de treino, um ruído aleatório vai gerar uma nova imagem de pessoa. Caso o ruído aleatório se mantenha constante, o modelo gerará sempre a mesma imagem.

Ao mesmo tempo, existe o discriminador que tenta descobrir se a entrada recebida pelo gerador é real ou falsa. Ambos os modelos trabalham um contra o outro, fíndando a competição entre ambos. O gerador tenta gerar uma saída tão realista que eventualmente engana o discriminador.

No geral, o objetivo do gerador é produzir exemplos falsos perfeitos que imitem as características do conjunto de dados de entrada. Os resultados devem ser tão perfeitos que seja difícil distinguir entre o original e o gerado. O gerador recebe o retorno das classificações do discriminador, e com esta informação os geradores sabem o que é possível alterar em suas amostras falsas para enganar o discriminador. Por outro lado, o objetivo do discriminador é classificar corretamente o que é real e falso, ou seja, ele também aprende com o desenvolvimento do gerador para que ele consiga detectar o máximo possível de amostras

falsas. A Figura 2.7, demonstra justamente o ciclo de aprendizado de uma GAN. No passo 1 da Figura 2.7 é dada uma entrada para o Discriminador. No passo 2, o gerador cria uma imagem aleatória partindo de um vetor aleatório. O Discriminador calcula as diferenças entre as imagens reais e falsas, passo 3, e informa o gerador para o mesmo melhorar suas imagens, passo 5. Esse resultado alimenta tanto o Discriminador quanto o Gerador no processo de treino descrito acima.

Figura 2.7: Arquitetura de funcionamento de uma GAN.

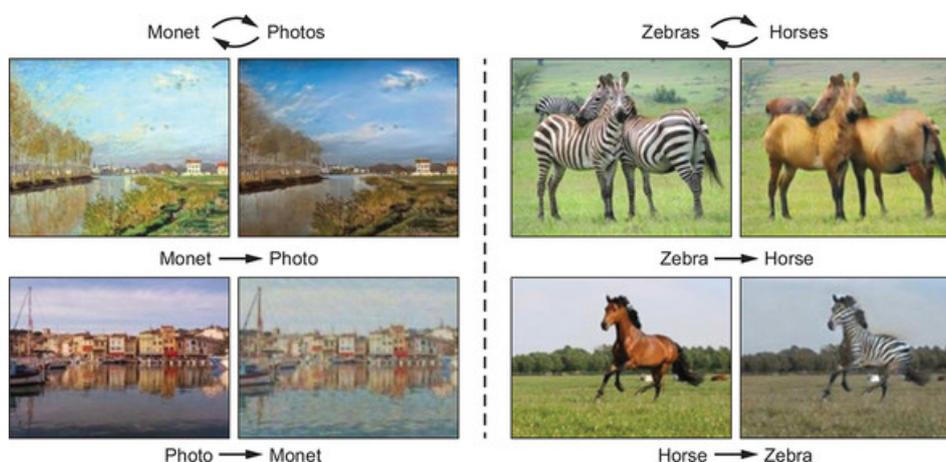


Fonte: (Langr e Bok 2019)

O gerador recebe um vetor de números aleatórios e o discriminador recebe exemplos falsos como entrada vindos do gerador e exemplos reais vindos do conjunto de dados de treinamento (Langr e Bok 2019). Portanto, ambos crescem ao mesmo tempo dentro desta competição, tornando o resultado tão perfeito que, na verdade, é imperceptível aos olhos humanos e apenas o discriminador consegue diferenciar. Os dois modelos devem chegar em um ponto tão avançado que o gerador não precisa mais do discriminador, ele precisa conseguir gerar imagens tão realistas com apenas um vetor de ruído aleatório que o discriminador não consiga mais detectar. Esse ponto avançado é quando uma GAN atinge um número específico de iterações, geralmente esse número é determinado quando o erro de validação é bom, mas depois de várias iterações começa a piorar.

GANs são utilizadas em casos onde o interesse é reproduzir algo com primor. De acordo com Yann Lecun, diretor de pesquisa de IA do *Facebook*, as GANs são a ideia mais legal em aprendizado profundo nos últimos 20 anos (Langr e Bok 2019). O fato de que GANs poderem recriar exemplos realistas e com qualidade muito boa é de fato único. Dentro de suas aplicações uma delas é possível traduzir imagens de um domínio para outro, por exemplo, transformar um cavalo em uma zebra ou até mesmo uma foto geral em uma pintura ao estilo do famoso pintor Claude Monet (Langr e Bok 2019).

Figura 2.8: Exemplo de tradução de imagens feitas por GANs.



Fonte: (Langr e Bok 2019)

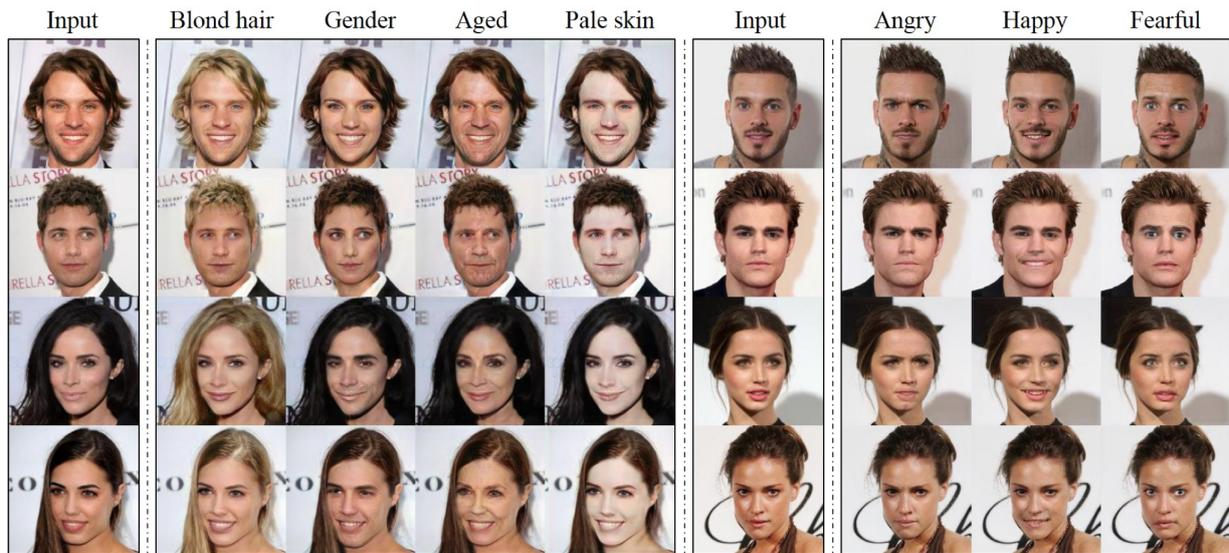
Na Figura 2.8 exemplifica como o processo de tradução de domínios funciona e como é um caminho bidirecional. Assim como na Figura 2.8, é possível obter imagens falsas de cavalos a partir de imagens originais de zebras, assim como é possível obter imagens falsas de zebras a partir de uma imagem de cavalo. O mesmo princípio é demonstrado para fotos de paisagens e pinturas, a partir de pinturas é possível obter uma fotografia falsa de uma paisagem e assim funciona o caminho invertido.

Estado da Arte em tradução de imagens por GANs

Desde então, a popularização das GANs está cada vez mais comum devido às suas aplicações. Hoje em dia, as mais populares são *StarGAN*(Choi et al. 2018), *StyleGAN*(Karras et al. 2020), *CycleGAN*(Zhu et al. 2017). Todas, em suas atuais versões, são consideradas o estado da arte em tradução de imagem feitas por GANs.

A *StarGAN*(Choi et al. 2018) surgiu devido a outras técnicas apresentarem muita dificuldade em tradução para multi-domínios. Então, com esta dificuldade, a *StarGAN* se tornou a base de outros projetos desse tema. Essa abordagem permite que uma única imagem seja traduzida em diversos outros domínios, assim como demonstrado na Figura 2.9 onde algumas entradas são traduzidas em domínios que mudam suas características fenotípicas e em outra abordagem, expressões faciais. Utilizando o banco de dados CelebA (Liu et al. 2015), a *StarGAN* realiza um treinamento simultâneo de vários conjuntos de dados com diferentes domínios em uma única rede. Fato que resulta em um melhor desempenho quando comparado com outras redes de literaturas anteriores.

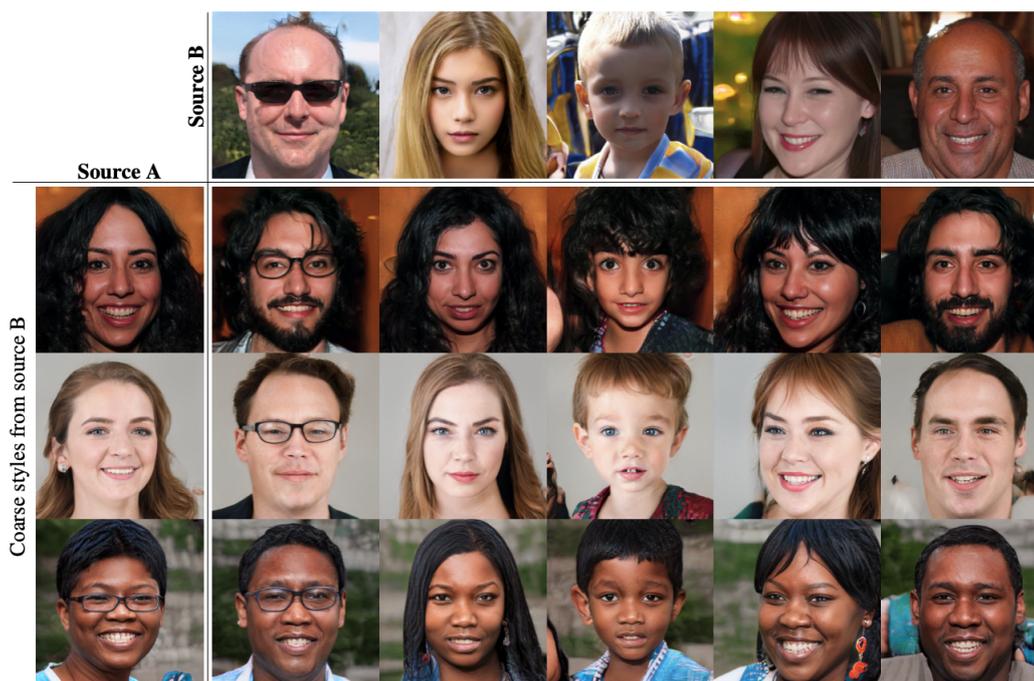
Figura 2.9: Tradução feitas pela *StarGAN*, iniciando com quatro entradas, é possível traduzir em diversos domínios sejam eles de características fenotípicas, sejam expressões faciais.



Fonte: (Choi et al. 2018)

StyleGAN foi primeiramente criada em 2018, essa rede possui uma arquitetura diferente que permite a criação de imagens. A segunda versão da *StyleGAN* é a rede que gera os rostos do site "Essa pessoa não existe" (Karras et al. 2019). A diferença entre uma GAN tradicional para a arquitetura da *StyleGAN* é a utilização de uma chamada rede de síntese, uma MLP na saída do espaço latente. Essa arquitetura permite que as imagens tenham uma alta qualidade de saída e consiga traduzir uma única entrada em diversas outras. A Figura 2.10 demonstra uma tradução de multi domínio feita pela *StyleGAN*. Com três imagens como entrada e outras cinco como outra, é possível cruzá-las e traduzir do domínio da primeira entrada para a segunda.

Figura 2.10: Tradução feitas pela StyleGAN de imagens de entrada (*Source A*) em um estilo das outras imagens *Source B*.



Fonte: (Karras et al. 2020)

Por último, a *CycleGAN* também propõe uma arquitetura diferente das GANs comuns. Enquanto a *StyleGAN* possui um espaço latente e uma rede de síntese para mapeamento de funções, a *CycleGAN* possui duas funções de mapeamento associadas a dois discriminadores. O primeiro discriminador incentiva o gerador a traduzir imagens dentro de um domínio do próximo discriminador para servir como entrada desse segundo discriminador. E esse ciclo vai se repetindo até a rede atingir o ponto desejado de saída. Para regularizar o mapeamento, J. Zhu et al. introduziram duas funções de perda chamadas *cycle consistency losses*. Essa técnica faz com que traduzindo de um domínio para outro e depois realizando o caminho inverso seja possível obter o mesmo domínio. Um bom exemplo é a Figura 2.8, onde de zebras, é possível traduzir a imagem para cavalos e a próxima imagem de cavalos para zebras.

2.2 REPRESENTAÇÃO HUMANA FACIAL

A Representação Humana Facial é um método onde envolve diversos processos que consigam traduzir características fenotípicas de um ser humano em uma representação por desenho ou edição de imagem. No Brasil, esse método é utilizado em casos de retrato-falado, identificação de vítimas de difícil reconhecimento e progressão de imagem de pessoas desaparecidas. Para todos os casos, é feito um levantamento a respeito dos possíveis dados da pessoa em questão. Por exemplo, em caso de desaparecimento, a roupa que estava utilizando, lugares que a pessoa costumava frequentar, onde a pessoa estava indo no momento

do desaparecimento.

Para executar um retrato-falado, a imagem final é baseada em depoimento de vítimas e testemunhas que auxiliem na constituição facial do agente em questão. Pode-se também utilizar imagens de câmeras e outros métodos digitais que auxiliem na identificação do autor de algum crime (Paulo). Assim como reconstituição da aparência de figuras históricas, é possível utilizar o mesmo método para reconhecimento de corpo de delito onde a identificação não foi possível ser feita por outros métodos. Sendo assim, é feito um estudo a partir das características dos ossos faciais, poupando custos de exames de DNA e tempo de investigação, por exemplo (Moraes et al. 2022).

Por outro lado, para a progressão de idade de pessoas desaparecidas, existem passos adicionais. Além do levantamento de dados, é feito um estudo a partir dos hábitos da pessoa através de uma entrevista com familiares. Fatos que possam influenciar na aparência da pessoa são extremamente importantes para a elaboração da progressão de idade e identificação, como hábitos de consumo alimentar, consumo de álcool, tabagismo, alguma enfermidade que altere a aparência. Após esse levantamento de dados e sobre a vida da pessoa e familiares, é feito um recolhimento de fotos da vítima. A última etapa é a manipulação artística no *software Photoshop*, essa manipulação é feita por um perito papiloscopista. Para essa manipulação, é feito um estudo sobre a morfologia e antropometria da pessoa desaparecida que seja fiel aos dados levantados e ao envelhecimento do núcleo familiar (Comunicação/DGPC).

De acordo com a perita papiloscopista Thalyta Oliveira citada em uma notícia da Polícia Civil do Distrito Federal (Comunicação/DGPC), não é possível obter um resultado exato da aparência atual de uma pessoa desaparecida com o trabalho de progressão de idade. Existem muitos fatores que influenciam na aparência de uma pessoa ao longo dos anos, adicionados ao trabalho da polícia no Brasil, pode-se tornar um processo ainda mais distante da realidade, logo a motivação deste trabalho. O uso de inteligência artificial pode tornar o processo do profissional papiloscopista menos laborioso ao trazer imagens base já feitas.

2.3 BANCOS DE IMAGENS FACIAIS

Existem vários bancos de dados focados em imagens do rosto de dezenas de pessoas. Portanto, esse tipo de banco é ideal para a prática deste trabalho. Os utilizados neste trabalho são *CelebA* (Liu et al. 2015), *UTKFace* (Zhang Zhifei e Qi 2017) e o de criação própria com base na informação disponível para o público. *CelebA* reúne mais de 200 mil imagens de rostos de celebridades com diferentes idades e diferenciados por mais de 40 atributos. *UTKFace* consiste em aproximadamente 20 mil imagens de rostos e também com marcação de atributos, como idade, cor dos cabelos, cor dos olhos, acessórios de rosto. Os detalhes a respeito do banco de dados criado manualmente serão tratados no Capítulo 4.

3 TRABALHOS RELACIONADOS

Apesar das GANs serem redes relativamente novas, a utilização das mesmas já está altamente disseminada. Há uma abundância de aplicações e resultados muito interessantes. M. Grimmer et al. realizaram um trabalho estilo *survey*, onde foi possível listar e comparar técnicas de progressão de idade usando DNN (Grimmer, Raghavendra e Busch 2021).

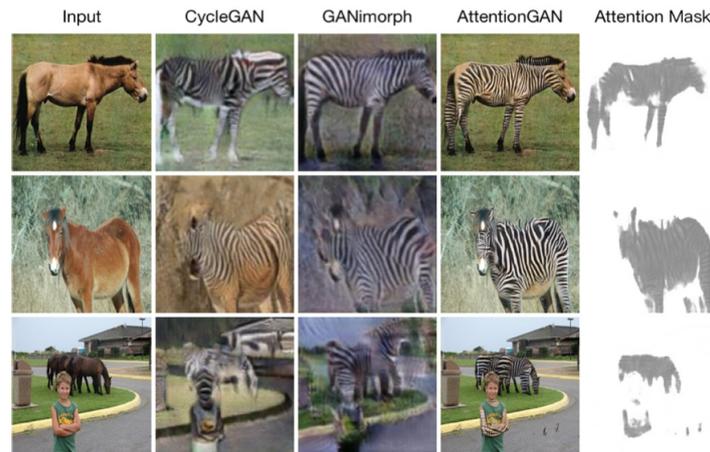
3.1 REDES ADVERSÁRIAS GENERATIVAS

Existem alguns trabalhos que utilizam apenas GANs para identificação ou progressão de idade, ou seja, se aproxima da proposta deste trabalho.

P.S. Chandran et al. propõem um sistema de identificação de crianças desaparecidas, este sistema pretende ser global, mas foi criado e testado com base no cenário de crianças desaparecidas na Índia (Chandran et al. 2018). Comparando com as técnicas no Brasil, este fluxo de trabalho proposto poderia substituir as duas primeiras etapas de coleta de dados sobre a pessoa desaparecida. Para a metodologia de pareamento de crianças, P. S. Chandran et al. usaram CNNs para extração de recursos de imagens e SVM como classificador para correspondência de faces. Um trabalho mais focado no sistema e em resultados de similaridade entre imagens de rostos. Os resultados obtidos pelos autores foi de 99,41% de precisão. No entanto, seu trabalho não especifica qual conjunto de dados foi usado para obter esses resultados, o que pode levantar questões sobre a diversidade desse banco de dados (Chandran et al. 2018). O Brasil é um dos países com maior diversidade étnica do mundo (Fisher). A diversidade étnica das imagens dos banco de dados é um tópico muito importante a ser considerado pois a rede a ser treinada aprende de acordo com aquelas características fenotípicas daquele banco.

Dentre outras abordagens existentes, a *AttentionGAN* é muito usada e colocada em comparação em trabalhos quando se fala de multi-tradução de imagens (Tang et al. 2021, Sharma, Sharma e Jindal 2021). H. Tang et al. utiliza uma arquitetura similar à *CycleGAN*, porém adiciona um recurso de máscara. Dessa maneira, é possível diferenciar o alvo da tradução de imagem do fundo da imagem unindo a máscara à tradução de saída e obter melhores imagens. Neste método, o discriminador considera apenas regiões importantes da imagem gerada. Essa abordagem foi capaz de demonstrar imagens mais precisas do que os outros modelos existentes, assim como demonstra a Figura 3.1. A última coluna da Figura 3.1 demonstra a demarcação dessas máscaras e compara os resultados obtidos com a *AttentionGAN*. As três técnicas receberam a mesma entrada, primeira coluna da Figura 3.1.

Figura 3.1: Comparação entre outros modelos de tradução de imagem e a *AttentionGAN* proposta por H. Tang et al.



Fonte: (Tang et al. 2019)

N. Sharma et al. apresentam uma abordagem combinada usando *AttentionGAN* (Sharma, Sharma e Jindal 2021) e SRGAN (Ledig et al. 2016). N. Sharma et al propõem uma arquitetura diferente, imagens pré processadas são dadas como entrada no gerador da *AttentionGAN*, nesse processo, o gerador consegue gerar as máscaras de atenção e finaliza seu teste. Após esse processo, existe um processo de decisão se precisa aplicar um filtro para melhorar a qualidade de pequenas áreas da imagem e após esse processo, as imagens, tanto as com filtro ou não passam pela *SRGAN*. Essa etapa final com a *SRGAN* tem o intuito de melhorar a qualidade e resolução das imagens. Fazendo todo esse processo, foi possível reduzir a complexidade computacional e o tempo de treinamento. N. Sharma et al. conseguiram gerar faces envelhecidas em super-resolução com alta preservação de identidade e com uma taxa de erro de 0,001% (Sharma, Sharma e Jindal 2021). Considerando esses resultados, a arquitetura proposta utilizando *AttentionGAN* é uma arquitetura interessante para estudar dentro do cenário brasileiro.

P.K. Chandaliya et al. utilizam *AttentionGAN* para envelhecer e rejuvenescer imagens de crianças. Neste trabalho, P. K. Chandaliya demonstra mais detalhes do envelhecimento em imagens infantis. Além disso, há um estudo para avaliar o quão fiel é a imagem gerada em relação à imagem original e demonstra que, como esse *framework* foi testado com um banco de dados de crianças, é possível obter resultados melhores do que outros trabalhos. (Chandaliya e Nain 2021)

V. Kamble e M. Dale tratam a respeito de identificação de recém-nascidos e crianças por meio de biometria facial utilizando CNNs (Kamble e Dale 2021). Neste trabalho, V. Kamble e M. Dale utiliza as técnicas de *Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA)* para extrair características fenotípicas dos rostos dessas crianças. Utilizando a técnica de extração de características, foi possível obter uma boa acurácia em reconhecimento facial. Porém, apesar dos bons resultados, V. Kamble e M. Dale citam que ainda é necessário testar a arquitetura proposta em um banco maior de imagens (Kamble e Dale 2021).

O cenário de identificação de pessoas apresentado por A. D. Rajkumar et al. (rajkumar R. Karthick Raja) é em áreas lotadas de pessoas, como festivais, ruas públicas. Os autores utilizam uma rede CNN para rea-

lizar essa tarefa de identificação. A estrutura proposta é primeiramente reunir imagens de pessoas desaparecidas em um único banco de dados, e é importante que o banco tenha várias fotos da mesma pessoa em diversos ângulos. Com o banco criado, A. D. et al utilizam a rede AlexNet (Krizhevsky, Sutskever e Hinton 2012) já pré treinada. Sendo assim, o dispositivo que captura as imagens desses locais públicos, como um *drone* por exemplo, envia as imagens para o que eles chamam de controlador. Neste controlador, as imagens dos rostos serão recortadas e inseridas em um banco de dados. Criado esse banco, o mesmo é utilizado como entrada para o treinamento da rede. Após o treino, existe o processo de classificação e comparação com o banco de desaparecidos (raj Kumar R. Karthick Raja).

Todos os trabalhos citados foram de extrema relevância para observação de como o assunto é tratado em outros países, quais os bancos de dados mais utilizados e também redes neurais e técnicas que podem ser relevantes para o cenário do Brasil.

3.2 RECONHECIMENTO FACIAL HUMANO

Como citado anteriormente, o Brasil utiliza técnicas conhecidas como reconhecimento facial humano em seu órgãos competentes de identificação humana. Diante desses fatos, C. L. F. de Alencar et al (Alencar et al. 2022) estudam como a Rede Integrada de Bancos de Perfis Genéticos (RIBPG) é eficaz frente a esses problemas. O RIBPG recebeu 70280 registros de perfis genéticos em 2019. Casos como o de Rachel Genofre RIBPG, uma menina desaparecida, foram solucionados graças ao sucesso do RIBPG. Assim, C. L. F. de Alencar et al reforçam a importância de campanhas de marketing que estimulem os familiares de pessoas desaparecidas a participar do banco de dados genético nacional (Alencar et al. 2022). A questão do RIBPG se assemelha muito à proposta deste trabalho e reforça a importância da unificação dos dados, imagens e progressão de idade de pessoas desaparecidas.

P. R. B. Reis e E. W. O. Filho investigam se as técnicas de composição facial aplicadas no Brasil são confiáveis e capazes de condenar um criminoso (REIS). Embora o contexto apresentado no trabalho seja para apurar a autoria de um crime, no Brasil, ele utiliza as mesmas técnicas de composição facial e progressão de idade para pessoas desaparecidas. Essa técnica é chamada de Reconhecimento Facial Humana. Os autores descrevem que em muitos lugares do Brasil essas técnicas são aplicadas de forma extremamente manuais. Composições faciais são desenhadas por especialistas em impressão digital com base no depoimento das vítimas. Em conclusão, os autores expõem o quanto é necessário atualizar essas técnicas para torná-las mais confiáveis e práticas (REIS). Os autores apresentam a lei brasileira de percussão quando ocorre um crime e técnicas para identificar o autor do crime. A técnica mais utilizada é o croqui policial, ou a chamada de Representação Facial Humana, que com o uso de softwares para composição, são utilizadas várias peças juntas para compor um único rosto. Porém, a confiabilidade desta técnica é questionada neste trabalho já que muitas vítimas podem ser traumatizadas ou produzidas muito tempo após o crime. Segundo P. R. B. Reis e E. W. O. Filho, na maioria dos Estados brasileiros, o croqui policial ainda é feito à mão e baseado na percepção que o policial teve de acordo com a vítima e montado com as peças fotográficas disponíveis no software (REIS) A questão principal do retrato falado policial apresentada neste artigo é relacionado a um problema denominado Falsas Memórias. Quando um fato é inserido por fatores externos ou falhas na reconstituição da memória, falsas memórias podem aparecer nos depoimentos,

principalmente de pessoas traumatizadas psicologicamente. A técnica de Reconhecimento Facial Humano quando utilizada para pessoas desaparecidas, é essencial o depoimento da família pois são coletados hábitos e fatos sobre a pessoa e a família que podem interferir no seu aparecimento (Comunicação/DGPC). Se a família está traumatizada por algum contexto do desaparecimento, a confiabilidade do processo cai; conseqüentemente, poderia interferir na escolha da fotografia para identificação e progressão da idade ao longo dos anos.

Ao contrário do trabalho anterior, no trabalho de A. Bragias et al. a tecnologia é utilizada para identificar criminosos pela polícia. Este trabalho foca na percepção do público sobre o uso de tecnologias para reforçar a identificação policial, especialmente o *Facial Recognition Technology* (FRT) (Bragias, Hine e Fleet 2021). De acordo com A. Bragias et al. , a percepção do público é proposta como outra variável para ter uma implementação FRT bem-sucedida, pois há questionamentos relacionados à privacidade das pessoas. Sua técnica para medir a percepção do público foi analisar os comentários da plataforma de vídeos e *streaming Youtube* em vídeos relacionados à implementação do FRT para questões policiais no período de 2017 a 2021. Implementando a análise sentimental, eles conseguiram identificar se a linha do comentário, ou seja, neutra, positiva ou negativo. Após muitas coletas de dados, o trabalho apresenta um conjunto de dados final com 609 comentários em 71 vídeos e 528 citações individuais dos autores (Bragias, Hine e Fleet 2021). A conclusão final deste trabalho é que a maioria dos comentários foram negativos sobre o tema. Apenas alguns positivos e neutros. Existem muitas preocupações sobre as pessoas relacionadas à aplicação do FRT pela polícia; primeiro, está relacionado à precisão da tecnologia. Em segundo lugar, os abusos de autoridade e, por último, a preocupação com os direitos e a liberdade das pessoas. De acordo com a pesquisa de A. Bragias et al. , as preocupações negativas são uma reação típica quando a polícia usa novas tecnologias; no entanto, muitas preocupações são replicadas virtualmente.

4 METODOLOGIA

Diante da motivação do trabalho, a fim atingir os resultados propostos, é fundamental seguir alguns passos para obter os dados brasileiros e posteriormente testá-los em diversas GANs e ferramentas. Primeiro passo é coletar dados entre vários estados brasileiros disponíveis publicamente. Uma vez obtidos o conjunto de dados, o segundo passo é a seleção de outros conjuntos de dados que mais se ajustassem ao problema. As características demográficas da população brasileira devem ser consideradas. Segundo o IBGE, a população brasileira é composta por 27,4% de pessoas de 0 a 19 anos. Entretanto, houve uma dificuldade de encontrar um banco de dados com esses requisitos de idade. Para os testes elementares das soluções propostas neste trabalho, foi utilizado o banco de dados CelebA (Liu et al. 2015). CelebA possui muitas imagens de pessoas desde sua infância até a fase adulta e é disponível publicamente. Por mais que este banco não ilustre a característica populacional que imprima as características do Brasil, os mesmos podem auxiliar e servir como prova de conceito.

4.1 ACESSO À INFORMAÇÃO

Há alguns anos, no Brasil, já existiam leis e medidas para prevenir e solucionar casos de desaparecimento de pessoas. Antes de 2019, cada estado brasileiro tinha sua própria forma de notificar desaparecimentos e suas diligências investigativas. Em 2019, foi criada uma lei federal para prevenir e investigar casos de pessoas desaparecidas de maneira única por todo o país. Dentro dessa Lei, além de criar uma única forma de investigação, ainda institui a criação de um banco de dados único onde todos os estados possuem acesso e também possam alimentar. No entanto, ainda não existe esse banco de dados integrado. Em 2022, foi feito um acordo onde foi aprovada a interoperabilidade entre bases de dados e dados de várias instituições responsáveis (Público). Portanto, no futuro, um banco de dados integrado com dados significativos deve estar disponível e acessível em todo o país. Existe um projeto de chamado Sistemas Nacional de Localização e Identificação de Desaparecidos (SINALID) (CNMP). Este projeto surgiu em 2017 para atender lacunas que o sistema de identificação brasileiro não estava. Esse projeto é gerido por cada estado, não é algo unificado, cada estado brasileiro é responsável por gerir e conceder acessos ao sistema. Existem muitas histórias de sucesso por conta do SINALID, porém, é um sistema restrito onde o público acessa somente em casos de registro de desaparecimento. Além de não disponibilizar imagens para o público, não foi possível obtê-las por meio da Lei de Acesso a Informação.

Considerando que as imagens e informações sobre desaparecimento de pessoas é um dado público, cabe acionar a Lei de Acesso à Informação (República), onde o público possui direito a acessar esses dados. Por meio do portal de transparência foi aberta uma requisição ao acesso ao banco de dados unificando referenciado na Lei 13.812/2019 (República). Entretanto, a resposta oficial recebida sobre essa requisição foi de que, infelizmente, ainda não existe esse banco de dados, por isso, não é possível acessá-lo. Entendendo este problema, foi necessário acessar as informações disponíveis *online* por cada polícia civil de cada estado e também acessar via aplicativo nacional chamado *SINESP Cidadão*.

O Sistema Nacional de Informação de Segurança Pública (SINESP), é um aplicativo disponível para *download* onde cidadãos conseguem acessar diversos tipos de informações. Informações como consulta de veículos, mandados de prisão e o mais importante, pessoas desaparecidas. Acessando o aplicativo, assim como na Figura 4.1, é possível visualizar e listar as pessoas desaparecidas, filtrar por categorias como idade, data, estado e cidade do ocorrido, dentre outras.

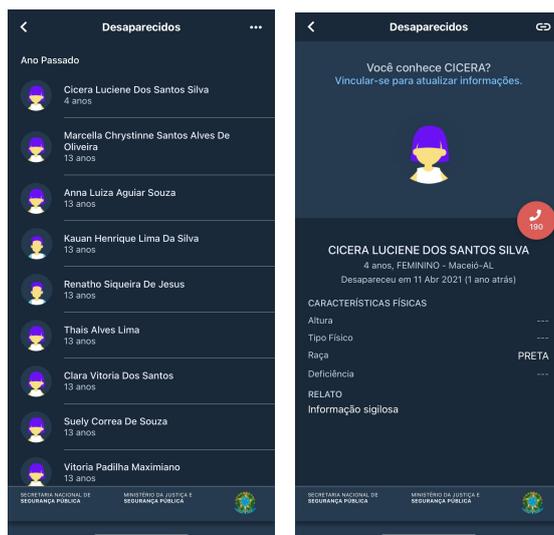
Figura 4.1: Sinesp Cidadão: Primeira tela do aplicativo onde é possível filtrar as informações das pessoas desaparecidas.



Fonte: Elaboração própria

Como demonstrado na Figura 4.2, clicando em um nome no aplicativo, é possível visualizar as pessoas desaparecidas individualmente como uma lista e pré visualizar apenas algumas informações. Nessa próxima tela, é possível observar campos como nome completo da pessoa, idade, sexo, etnia e dados do fato. Mesmo que haja informações sobre cidade e estado, mas não é possível determinar se a pessoa nasceu ou desapareceu neste local. Além disso, não há informações sobre o contexto do desaparecimento, ou seja, roupas usadas no dia fatídico, o local específico onde a pessoa estava localizada, o que estava fazendo, características físicas e o mais importante, a fotografia da pessoa. Nenhuma das pessoas na lista do aplicativo possui uma fotografia. Além disso, há poucos nomes na lista, o que pode indicar que há uma inconsistência nesse banco de dados utilizado pelo SINESP Cidadão. Por este motivo, a base de dados do SINESP não pôde fornecer os dados necessários para este trabalho.

Figura 4.2: Sinesp Cidadão: Lista de pessoas disponível no aplicativo e exemplo do detalhamento de uma delas



Fonte: Elaboração própria.

Determinado que a base de dados do aplicativo governamental não seria utilizado neste trabalho, foi feita uma análise das informações disponíveis por cada estado brasileiro para construir um banco de dados próprio.

A Tabela 4.1 demonstra as fontes e as informações disponíveis para cada possível local de consulta. Foram considerados o Registro Nacional de Pessoas Desaparecidas, banco institucionalizado pela Lei 13.812/19 (República), aplicativo SINESP Cidadão, fontes de segurança pública de cada estado, como *disk denúncia* e site das Polícias Civas. Foi possível observar que em muitos estados as informações não estão disponíveis em nenhum canal aberto. Estados como Acre, Amapá, Paraíba, Piauí e Rondônia. Em outros, existem informações que aparentemente estão inconsistentes pois faltam o registro de algumas pessoas, como Maranhão, Roraima e Sergipe. No Pará, existe apenas uma fonte onde divulga apenas menores de 18 anos desaparecidos, para adultos não existe fonte de consulta disponível. Para o Rio Grande do Sul, existem muitos registros sem imagens no *website*. Para o Ceará, Mato Grosso e Tocantins, a polícia utiliza das redes sociais para divulgação de imagens e informações das pessoas que desaparecem e o status de seu desaparecimento. Fontes que possuem dados inconsistentes não foram consideradas para montar o banco de dados pois ou os dados realmente não estão abertamente disponíveis ou não estão aptos para construir o banco. Para estados onde aparentemente existem poucas pessoas divulgadas foram considerados, porém é um ponto de inconsistência pois nem todas as imagens e informações de pessoas nesse estado de desaparecimento estão disponíveis. Estados com dados disponíveis e considerados para a construção do banco de imagens são Amazonas, Bahia, Distrito Federal, Espírito Santos, Goiás, Maranhão, Minas Gerais, Pará, Paraná, Pernambuco, Rio de Janeiro, Rio Grande do Sul, Sergipe, Santa Catarina e São Paulo. Apenas Distrito Federal possui Progressão de idade para todos os desaparecidos que se encaixam contexto da Lei para elaboração de progressão de idade. Paraná e Santa Catarina possuem algumas imagens de progressão de imagem disponíveis.

Tabela 4.1: Fontes de dados de cada estado Brasileiro

Fonte	Local de consulta
Registro Nacional de Pessoas Desaparecidas	Lei 13.812 (República)
SINESP Cidadão	Aplicativo SINESP Cidadão (Pública 2020)
Acre	Participante do Projeto SINALID (CNMP)
Alagoas	Redes sociais do órgão policial (PLID/AL)
Amapá	Participante do Projeto SINALID (CNMP)
Amazonas	Site oficial do órgão policial (Amazonas)
Bahia	Site oficial disk denuncia do estado (Pessoa)
Ceará	Redes sociais do órgão policial (Ceará)
Distrito Federal	Site oficial do órgão policial (Comunicação/DGPC)
Espirito Santo	Site oficial disk denuncia do estado ((SESP))
Goiás	Site oficial do governo do estado (EXPRESSO)
Maranhão	Site oficial disk denuncia do estado (Maranhão)
Mato Grosso	Site oficial do órgão policial e redes sociais (Grosso)
Mato Grosso do Sul	Site oficial do órgão policial e redes sociais (Sul)(Sul)
Minas Gerais	Site oficial do órgão policial (Gerais)
Pará	Site oficial do órgão policial (Pará)
Paraíba	Participante do Projeto SINALID (CNMP)
Paraná	Site oficial do órgão policial (Paraná)
Pernambuco	Site oficial do órgão policial (Pernambuco)(Pernambuco)
Piauí	Participante do Projeto SINALID (CNMP)
Rio de Janeiro	Site oficial do órgão policial (Janeiro)
Rio Grande do Norte	Site oficial do órgão policial (Norte)
Rio Grande do Sul	Site oficial do órgão policial (Sul)
Rondônia	Participante do Projeto SINALID (CNMP)
Roraima	Site oficial do órgão policial (Roraima)
Santa Catarina	Site oficial do órgão policial (Catarina)(Catarina)
São Paulo	Site oficial da secretaria de segurança pública do estado (Paulo)
Sergipe	Site oficial do órgão policial (Sergipe)
Tocantins	Redes sociais do órgão policial (TO)

4.2 EXTRAÇÃO MANUAL E AUTOMÁTICA DAS IMAGENS

Na construção do banco de imagens com brasileiros, foi utilizado dois métodos. O primeiro deles foi um método totalmente manual e o segundo foi utilizando um código em linguagem *Python* para extração automática dos sites policiais.

Primeiramente, o método manual foi escolhido pois foi necessário observar alguns quesitos das imagens. Por exemplo, qualidade da foto original da pessoa desaparecida, se aquela pessoa possuía uma progressão de imagem e as informações que acompanham aquela imagem a respeito do contexto de desaparecimento. Esse processo manual foi feito apenas para os estados Bahia, Distrito Federal, Paraná e Rio de Janeiro. Como estes dados disponibilizam imagens de Progressão de idade, foram escolhidos para essa coleta manual e a análise de qualidade de imagem e informação disponibilizada. Bahia e Rio de Janeiro foram coletados manualmente também apesar não disponibilizarem progressão de idade, mas já adicionariam volume para o banco de dados. Nessa coleta manual, os portais das polícias civis de cada estado foram acessados, em sua maioria, através do campo de desaparecidos. As avaliações e informações sobre as imagens foram feitas uma a uma e planilhadas. Essa informação está disponível no Apêndice 1, 2, 3 e 4. A motivação dessa avaliação é para entender se as imagens estão apropriadas para um treinamento e validação de uma rede neural. Adicionalmente, as imagens receberam uma classificação de idade de desaparecimento, quando se aplica, a idade da progressão de idade. essa classificação e listagem das imagens pode ser utilizada para trabalhos futuros utilizando o banco por meio de uma escala própria. Além disso, foi estabelecido um padrão de nomeação para salvar essas imagens. Inspiradas no conjunto de dados CelebA(Liu et al. 2015), as imagens foram nomeadas de acordo com a rotulagem lógica do CelebA(Liu et al. 2015). Exemplificando, as imagens foram nomeadas da seguinte forma: primeiro uma sequência de números, ou seja, 01, 02, etc. Segundo, a letra A em caixa alta seguida da idade em que a pessoa desapareceu. Por último, se for uma progressão de idade, a idade seria substituída pela idade avançada seguida da letra P em caixa alta. Como exemplo, a terceira pessoa apresentada no site da polícia Paranaense foi Ciclano Silva que desapareceu quando tinha 7 anos e sua imagem progredida seria dele com 19 anos. Então, a imagem original de Ciclano Silva será rotulada como 003A07 e sua imagem progredida será rotulada como 003A19P. Esse padrão foi extremamente importante para separar os dados e dividir as faixas etárias por crianças (0 a 18 anos), adultos e idosos (a partir dos 60 anos) no treinamento dos dados. Definido esse padrão, todos os sites foram acessados e as imagens salvas manualmente em uma pasta exclusiva.

Por outro lado, para a extração automática das imagens foi elaborado um código em *Python* disponível no Apêndice I.2 onde o principal ponto foi a biblioteca chamada *BeautifulSoup*(Richardson 2007). Essa biblioteca serve para extrair dados de páginas HTML e XML. Seu funcionamento se baseia em varredura de árvores de uma página HTML. É possível identificar, navegar e extrair itens como imagens em uma página *web*. Além da *BeautifulSoup*, foi utilizada a biblioteca *Selenium* ~ (Muthukadan 2018). Esta biblioteca é destinada para testes em desenvolvimento *web*, porém, é utilizada neste trabalho para navegar nos portais como um usuário comum. Por exemplo, quando uma fonte possui paginação, com o *Selenium* é possível mapear e clicar na próxima página. Com ambas as bibliotecas implementadas, se navega pela página e identificadas pela árvore HTML, se extrai uma imagem.

Este código foi utilizado para identificar as imagens dos portais de polícia e salvá-las em um diretório

identificado. O código é estruturado em funções para cada estado, visto que cada fonte possui um desenvolvimento diferente. E recebe as entradas onde seriam os páginas das Polícias Civas interessantes ao trabalho. Com as fontes anteriormente mapeadas e automatizada a extração dessa maneira, foi possível aumentar o volume do banco de dados proposto neste trabalho.

A única ressalva dessa etapa é de não analisar a qualidade da imagem, sendo necessário, futuramente, esse mapeamento e listagem das imagens salvas pela execução do código manualmente. Não é essencial para a continuidade dos objetivos propostos neste trabalho, mas seria interessante para trabalhos futuros manipularem as imagens e informações do banco. As fontes que disponibilizam em seus portais e onde utilizam redes sociais também é possível utilizar o código para extração automática, mas devido a imagem ser editada pela polícia com as informações, seria necessário uma edição dessas imagens manualmente assim como demonstra a Figura 4.3. Por exemplo, os estados de Alagoas, Ceará, Mato Grosso do Sul, Santa Catarina (Catarina), Roraima e Tocantins.

Figura 4.3: Comparação entre uma imagem disponibilizada pela Polícia Civil do Amazonas (Amazonas) e uma disponibilizada pelo SINALID Alagoas (PLID/AL).



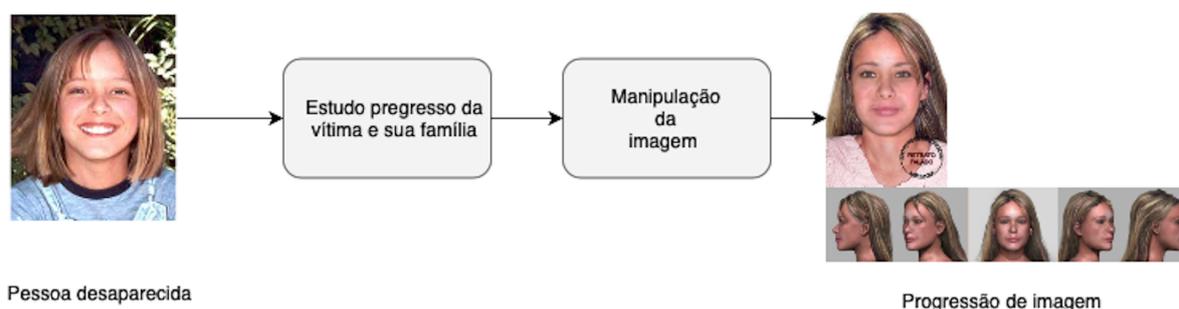
Fonte: Elaboração própria.

Ambas as situações foram consideradas para compor o banco de dados, apesar das imagens do estado de Alagoas e os outros exemplificados acima possuírem elementos na imagem, a fotografia da pessoa desaparecida ainda é uma fotografia de boa qualidade e com um ângulo claro que dê para as redes neurais trabalharem. Adicionalmente, para as imagens do estado do Pernambuco, também é necessária uma intervenção manual pois a polícia disponibiliza o cartaz de desaparecido em formato *pdf*. Mesmo que o código seja editado para extrair o *pdf*, ainda seria necessário entrar manualmente e extrair as imagens. Fatos que ressaltam a importância da criação do banco nacional unificado onde todas as imagens poderiam seguir um padrão e facilitar a troca de informações, aplicação de novas tecnologias e auxiliar em questão de precisão nas imagens de progressão.

4.3 ARQUITETURA PROPOSTA

Atualmente, o processo de progressão de idade utilizado pelas autoridades competentes segue uma linha do que já foi abordado. Como representado na Figura 4.4, primeiro são recolhidas informações a respeito da pessoa desaparecida incluindo sua imagem principal a ser divulgada. Passados uns anos, é feito um estudo de sua fisionomia, hábitos e progresso familiar, é feita uma manipulação utilizando *Photoshop* para envelhecer aquela pessoa de acordo com o tempo ocorrido. Na Figura 4.4, a criança em questão possui 10 anos e sua progressão seria a mesma com 28 anos em 2018 quando a progressão foi feita.

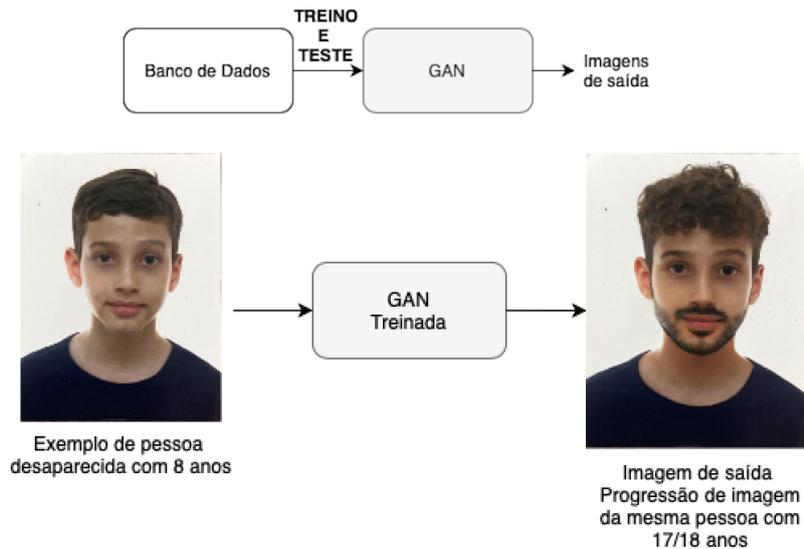
Figura 4.4: Exemplo do processo utilizado pelas Polícias Civis. Imagens reais retiradas da Polícia Civil do Distrito Federal.



Fonte: (Comunicação/DGPC)

A arquitetura proposta por este trabalho está representada na Figura 4.5. Onde é possível observar que primeiramente será utilizado o banco de dados unificado como entrada para treino, teste e validação de uma rede neural do tipo GAN. Após essa etapa de treino, as imagens de saída serão as imagens de progressão assim como no esquema abaixo representado na Figura 4.5. Após a GAN ser propriamente treinada e atingindo o ponto máximo de aprendizado, ou seja, quando a rede chega em um ponto de aperfeiçoamento que não aprende mais. Será possível entregar uma imagem para a Rede e ela envelhecer essa imagem de acordo com a idade da pessoa desaparecida. Ou seja, se a pessoa desapareceu com 8 anos em 2013, pode envelhecê-la para a idade de aproximadamente 18 anos em 2023.

Figura 4.5: Arquitetura proposta utilizando um banco de dados unificado e algoritmos de aprendizado de máquina.



Fonte: Elaboração própria.

4.4 AMBIENTE DE SIMULAÇÃO

Com o banco de dados construído, a próxima etapa é ajustar os ambientes de simulação para iniciar os treinos e validações das GANs a serem utilizadas. As simulações com a AttentionGAN foram feitas em um ambiente containerizado *Docker* (Merkel 2014). A máquina que abriga este ambiente possui um sistema operacional *Ubuntu 18.04 LTS* (Ubuntu releases 2018). Adicionalmente, a imagem do contêiner de simulação já dispõe do *CUDA Toolkit* (Developer 2023) e software como *Anaconda* (Anaconda Software Distribution 2020), bibliotecas do *Python*, como a *Tensorflow* (Abadi et al. 2016) e *OpenCV* (Bradski 2000). As dependências exigidas pela AttentionGAN e StarGAN seriam o *Python*, *TensorFlow* e uma biblioteca chamada *Pytorch* (Paszke et al. 2019).

Além da máquina de simulação, foi utilizada a ferramenta *Google Colab* (Bisong 2019) para executar o projeto com a CycleGAN. A única modificação necessária na plataforma é habilitar a utilização de GPU.

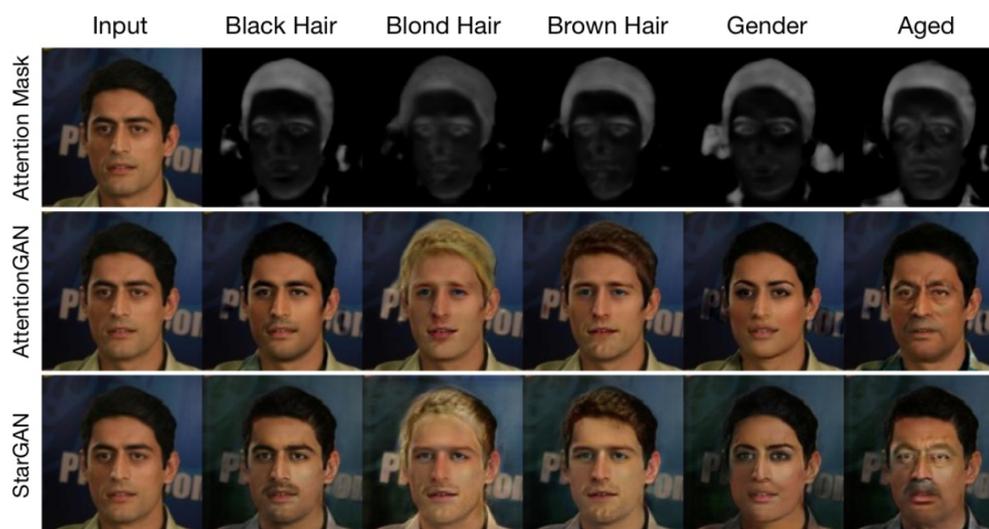
Para utilizar o *FaceApp* (Goncharov 2017), apenas um telefone móvel com sistema operacional iOS foi utilizado juntamente com a aplicação disponível online, *Face++* (Face++ 2017).

AttentionGAN

A principal referência de AttentionGAN é o trabalho de L. Hao et al. (Tang et al. 2021). Acessando seu repositório, a parte utilizada é a AttentionGAN-v1-multi. A AttentionGAN-v1-multi é uma implementação utilizando *PyTorch* para a tradução de imagens em diversos outros domínios, esse em específico é para tradução de rostos em outros domínios. Por exemplo, cabelos loiros em escuros, pessoa jovem em uma pessoa mais velha, homem em mulher.

L. Hao et al. adicionam as chamadas máscaras de atenção à StarGAN (Choi et al. 2018), resultando em seu projeto AttentionGAN-v1-multi. A StarGAN pode traduzir com flexibilidade uma imagem de entrada para qualquer domínio de destino desejado usando apenas um único gerador e um discriminador. Assim como descrito no capítulo 2, esse método combinado resulta em imagens mais precisas. Na Figura 4.6 é possível observar que ao utilizar uma *Attention Mask* a tradução fica mais precisa pois respeita os limites da imagem de entrada.

Figura 4.6: Comparação entre os resultados de tradução de imagem em diversos domínios feitos pela AttentionGAN e StarGAN.



Fonte: (Tang et al. 2021)

Após configurado o ambiente de simulação, foram seguidos os passos disponíveis no repositório do trabalho de L. Hao et al. O repositório foi clonado no contêiner de simulação e foi criado um ambiente *conda* para isolar os pacotes necessários para o projeto. Por exemplo, é descrito por L. Hao et al. que é necessário um *Python* versão 3.6.9 ou superior, portanto, foi mais fácil criar esse ambiente *conda* de execução. Primeiramente foi utilizando o banco de dados CelebA e seguindo os passos de treino e teste, foi possível obter imagens semelhantes às demonstradas por L. Hao et al. como mostra a Figura 4.7. O código do projeto (Tang et al. 2021) concentra diversas imagens em apenas uma e trabalha em cada uma delas, essa configuração de imagens por cada grupo e tamanho da imagem são itens modificáveis.

Figura 4.7: Resultados prévios obtidos durante a metade do treino da AttentionGAN seguindo os passos disponíveis no repositório do trabalho de L. Hao et al.

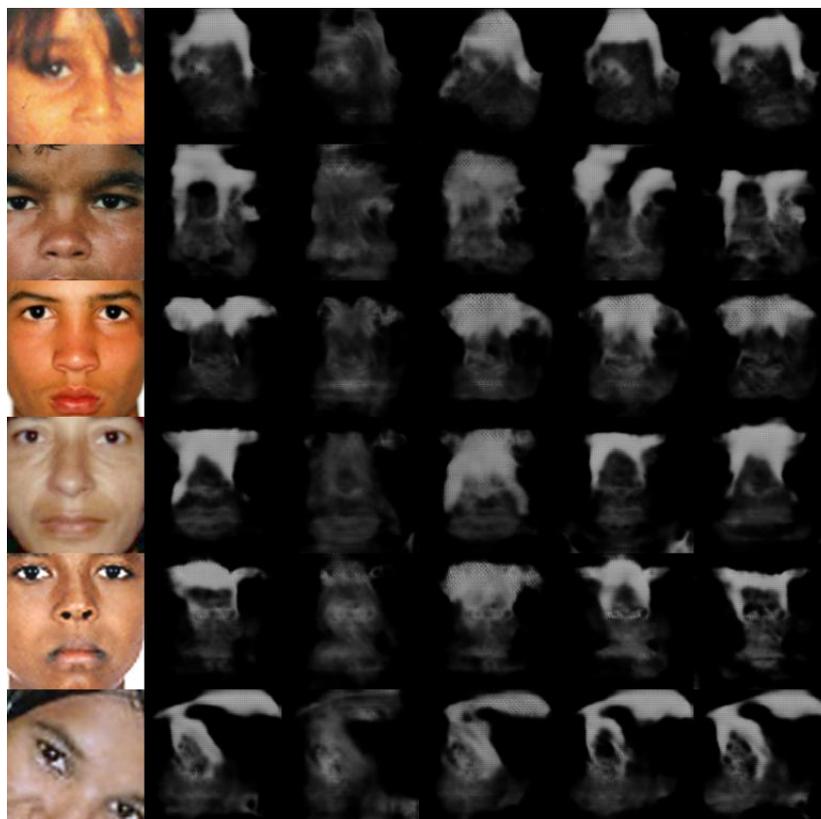


Fonte: Elaboração própria

Após realizados alguns testes com o banco CelebA, foi utilizado um pequeno grupo de imagens do banco de dados construído, no caso da Figura 4.8, são 6 imagens do estado do Distrito Federal. Esse teste foi realizado de forma reduzida por conta do vetor de atributos das imagens. Esse vetor de atributos define cada característica de cada imagem, o motivo de utilização desse vetor é uma exigência da arquitetura da StarGAN. Esse vetor é um conjunto de dados onde especifica cada imagem e suas características únicas. Por exemplo, caso a imagem 01 seja de uma pessoa de cabelos escuros e barba, no campo de cabelos escuros e barba, estarão os valores de 1, o que a imagem não conter de características recebe o valor de -1. No total são 37 características mapeadas para cada imagem do banco CelebA. Esse vetor está pronto e é disponibilizado junto com o banco de dados CelebA e foi utilizado no treino, teste e validação anterior.

As pessoas desaparecidas escolhidas para este pequeno teste foram utilizadas pois todas representadas na Figura 4.8 possuem progressão de imagem. Uma vez feito o treino e o teste com parte dessas 6 imagens, seria possível observar e comparar a tradução das imagens e sua progressão de imagem disponibilizada pela Polícia Civil do Distrito Federal. O vetor de características foi um grande empecilho neste trabalho, visto que não foi possível extrair essas características por forma automática, por exemplo, por meio de outra Rede Neural. Manualmente também não foi possível visto que são muitas imagens para realizar esse tipo de tarefa onde é necessário abrir cada uma delas e preencher o vetor de entrada de cada uma delas. Além disso, problemas com a arquitetura da AttentionGAN devido a dimensão de treino e testes, não foi possível finalizar e validar com as imagens do Distrito Federal. Como a AttentionGAN utiliza cerca de 202.599 imagens onde 80% delas são destinadas para treino o restante teste e validação, não foi possível realizar o mesmo processo com uma pequena amostra do banco. As ideias para contornar tais empecilhos estão descritas no Capítulo 6. Na Figura 4.8 é possível observar as máscaras de atenção sendo criadas no início do treino da rede.

Figura 4.8: Início dos treinos da AttentionGAN com as imagens da Polícia Civil do Distrito Federal.



Fonte: Elaboração própria

CycleGAN

Inspirado no aplicativo *FaceApp* (Goncharov 2017), A. Chandra propõe uma arquitetura de tradução de imagem para envelhecimento de imagens. A principal proposta é utilizar a Rede Neural CycleGAN com o banco de dados UTKFace (Zhang Zhifei e Qi 2017) para converter imagens de pessoas entre 20 e 30 anos

em aproximadamente 50 a 60 anos. São feitas algumas modificações no próprio código da rede neural e disponibilizado formato *Jupyter notebook*. Os resultados desta arquitetura está demonstrada na Figura 4.9. Acima é a imagem de exemplo original e abaixo a imagem envelhecida utilizando o projeto proposto.

Figura 4.9: Resultados de envelhecimento obtidos por A. Chandra em seu artigo.



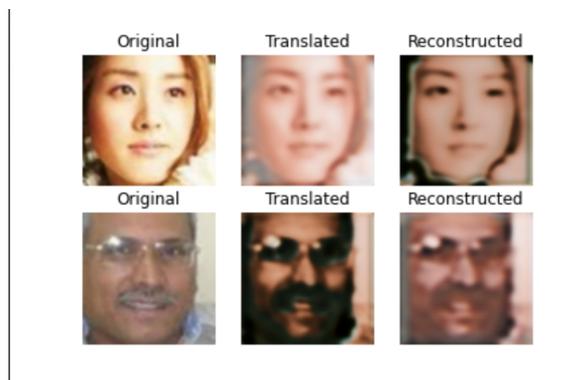
Fonte: (Matcha 2019)

Comparativamente, os resultados utilizando CycleGAN, assim como no trabalho de A. Chandra (Matcha 2019), foi possível obter resultados similares utilizando o banco de dados UTKFace. Porém, visivelmente os resultados não são tão precisos quanto os apresentados pela AttentionGAN-v1-multi. O que leva a supor que a utilização da técnica de máscaras de atenção, juntamente com a StarGAN, ainda é uma das melhores técnicas da literatura.

Em questões de execução, não foi possível executar o projeto por 10 épocas ou mais assim como o artigo recomenda. As épocas seriam as vezes que as entradas são exibidas para a rede, nesse caso, são 10 vezes. O projeto original foi feito na plataforma *Jupyter* e está no formato de *Jupyter notebook* e o recomendável é utilizá-lo na plataforma *Google Colab*. Porém, essa ferramenta impõe uma limitação de utilização de *GPU*, portanto, o treino não consegue se completar. O máximo de épocas que foi possível executar foram cinco épocas e os resultados obtidos utilizando UTKFace estão representados na Figura 4.10. Os resultados obtidos utilizando o banco de dados Brasileiro deste trabalho se apresenta na Figura 4.11. Para a Figura 4.10, foram apenas cinco épocas e o resultado ainda não foi o esperado como apresen-

tado na Figura 4.9, não há características de envelhecimento nas imagens reconstruídas e utiliza o banco UTKFace. A Figura 4.11 demonstra as imagens ainda no início do treino, mas sem resultados significantes para envelhecimento da imagem original.

Figura 4.10: Resultados reproduzidos do projeto de A. Chandra (Matcha 2019) utilizando o UTKFace.



Fonte: Elaboração própria

Figura 4.11: Resultados reproduzidos do projeto de A. Chandra (Matcha 2019) utilizando o banco de dados Brasileiro manualmente construído.



Fonte: Elaboração própria

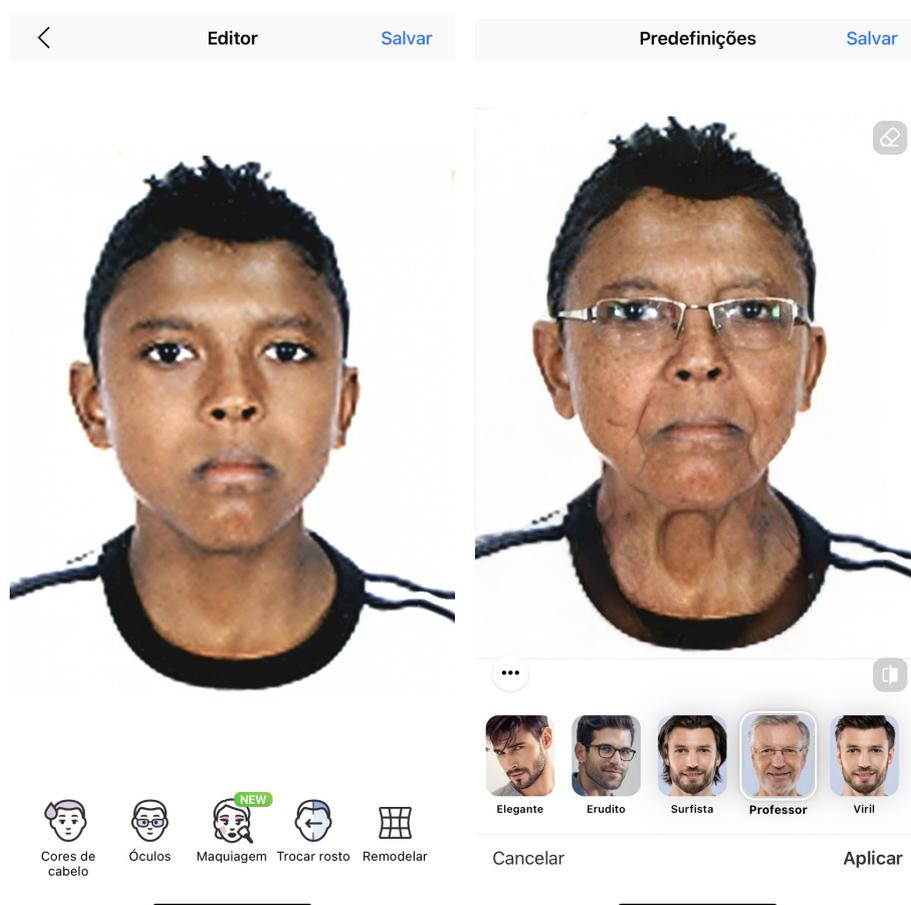
FACEAPP E FACE++

Y. Goncharov é responsável pelo desenvolvimento da inteligência artificial por trás do aplicativo Face-App (Goncharov 2017). Disponível apenas para dispositivos móveis, o aplicativo visa entregar edição de fotos de acordo com o reconhecimento facial. Apesar de muitos recursos serem pagos, o aplicativo possui uma poderosa rede neural tanto para reconhecimento facial quanto para realizar a aplicação dos filtros nas imagens. Por ser uma rede já treinada e disponibilizada em formato aplicativo, não é necessário utilizar um conjunto de dados para treino e teste. Apesar de não ser possível uma edição em massa, o aplicativo

fornece muitas modificações de imagens, como idade, barba e maquiagem. Aplicando essas modificações, é possível salvar uma nova imagem no dispositivo. Neste trabalho, foram utilizadas imagens do Distrito Federal extraído do banco de dados construído e imagens de crianças e adolescentes não desaparecidos. A razão de utilizar fotos de pessoas que não desapareceram é justamente para validação. Esse passo de validação seria incluído para as simulações anteriores mas que foi possível devido a impedimentos técnicos. Para essa pessoa não desaparecida, foram coletadas duas fotos com diferentes idades, 11 e 21 anos. As pessoas selecionadas do banco possuem 10 anos e 15 anos, suas imagens progredidas são com 28 anos.

O aplicativo possui diversos menus com modificações feitas manualmente em detalhes mínimos de acordo com cada menu como à esquerda da Figura 4.12, ou estilos pré definidos como demonstrado à direita na Figura 4.12. É possível observar como as modificações de idade se atentam à detalhes muito comuns do envelhecimento humano, como a dobra nasolabial, área do pescoço, queixo e boca.

Figura 4.12: Menus e modificações disponíveis no aplicativo *FaceApp*.



Fonte: Elaboração própria

Para modificação das pessoas desaparecidas e gerar suas imagens progredidas foram utilizando os menus de Cabelo, Gênero e Tamanhos.

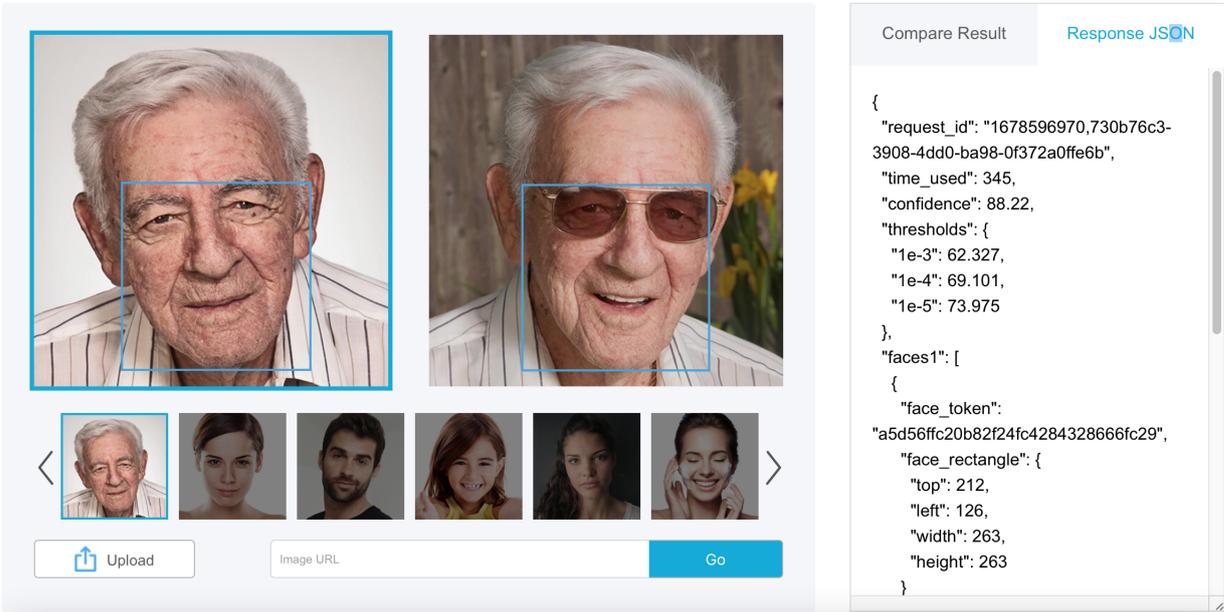
A aplicação *Face++* (Face++ 2017) é um projeto onde concentra diversas ferramentas relacionadas a reconhecimento de identidade feitas por *Deep Learning*. Tais ferramentas podem ser integradas via *API*

para diversas soluções empresariais. Neste trabalho, foi utilizada a ferramenta de Comparação Facial. Esta ferramenta também é possível ser utilizada via *API*, mas neste contexto, foi feito o carregamento das imagens elaboradas no *FaceApp* juntamente com suas progressões de idade elaboradas pela Polícia Civil do Distrito Federal e comparadas. Quanto maior o número de acurácia apresentado pela *Face++*, maior o nível de similaridade. Na Figura 4.13, em uma versão *demo* da Comparação Facial, em uma comparação entre duas imagens, a ferramenta retorna tanto a mensagem de probabilidade de ser ou não e o valor da acurácia dessa análise. No caso da Figura 4.13, o nível de similaridade entre as duas imagens de exemplo é o valor de 88.22, ou seja, um valor bem alto indicando que fortemente provável que seja a mesma pessoa.

Figura 4.13: Ferramenta de Comparação Facial disponível em Face++ (Face++ 2017)

Demo

Try Face Comparing now by uploading local images, or providing image URLs.
This demo is built with Search API and FaceSet APIs. If you have any specific technical requirements, check the index below or contact us.



```
Compare Result Response JSON  
{  
  "request_id": "1678596970,730b76c3-3908-4dd0-ba98-0f372a0ffe6b",  
  "time_used": 345,  
  "confidence": 88.22,  
  "thresholds": {  
    "1e-3": 62.327,  
    "1e-4": 69.101,  
    "1e-5": 73.975  
  },  
  "faces1": [  
    {  
      "face_token":  
      "a5d56ffc20b82f24fc4284328666fc29",  
      "face_rectangle": {  
        "top": 212,  
        "left": 126,  
        "width": 263,  
        "height": 263  
      }  
    }  
  ]  
}
```

Fonte: Elaboração própria

5 RESULTADOS

Neste capítulo, é mostrado como que os dados disponíveis pelas forças policiais brasileiras foram usados na proposta de arquitetura do processo de progressão de idade. Foram feitas simulações usando três diferentes abordagens. Por fim, os resultados são comparados e analisados em vista do cenário atual do Brasil nas tratativas de progressão de idade. Foram reunidos os dados disponíveis pelas forças policiais brasileiras, confeccionado uma arquitetura do processo de progressão de idade e feitas simulações utilizando três diferentes abordagens, foi possível reunir os resultados e compará-los com o cenário atual do Brasil nas tratativas de progressão de idade.

5.1 DISPONIBILIZAÇÃO DOS DADOS E QUALIDADE DE INFORMAÇÕES

Como demonstrado antes, apesar da existência da Lei que regulamenta o Registro Nacional de Pessoas Desaparecidas (República) e o projeto SINALID (CNMP). Os dados não estão reunidos e nem disponibilizados para consulta geral do público. Existem outras ferramentas disponibilizadas pelo Governo Brasileiro como o SINESP Cidadão, mas com dados inconsistentes e sem imagens. O que evidencia a necessidade da criação de um banco unificado e nacional. Neste trabalho, foi possível reunir as fontes disponíveis e classificá-las em relação à disponibilidade da informação e à qualidade desta informação exposta publicamente.

A Tabela 5.1 demonstra a comparação entre outras fontes e as informações disponíveis. Fontes onde possuem apenas a foto original das pessoas desaparecidas foram consideradas como aptas para treino das GANs. Fontes que imagens de progressão de idade, foram consideradas como aptas para treino e teste de GANs. O critério de nota para os dados disponibilizados por cada força policial foi criado de uma maneira em que a nota oscile entre zero e cinco. Uma nota zero significa que a fonte não possui os dados centralizados, não existem ou não estão publicamente disponíveis, como mostrado na Tabela 5.1, o Registro Nacional de Pessoas Desaparecidas se encaixaria nessa nota. Uma nota um seria quando as informações não estão devidamente divulgadas, dificultando o acesso. Um exemplo de nota um seria o estado do Acre onde foi possível encontrar que o estado faz parte do programa do Sistema Nacional de Localização e Identificação (CNMP), mas o estado não divulga em seus portais oficiais dados sobre desaparecimentos do estado. Nota dois seria poucas informações disponíveis e sem imagens, por exemplo, o aplicativo SINESP Cidadão. Para nota três seriam fontes que disponibilizam imagens originais, idade da pessoa desaparecida e local do ocorrido, por exemplo, estado da Bahia. A nota três foi considerada para estados que apesar de não divulgarem em portais oficiais, alimentam redes sociais onde disponibilizam imagens e contexto de desaparecimento. A próxima nota seria a quatro, direcionada a portais que disponibilizam imagens originais, informações sobre idade, data e local de desaparecimento e para alguns casos imagens de progressão de idade. Paraná e Santa Catarina receberam nota quatro pois alguns casos em que é cabível a progressão de idade não existem as imagens. Por fim, nota cinco é para fontes que disponibilizam imagens originais, quando cabível, todas as pessoas possuem progressão de idade e informações a respeito da pessoa

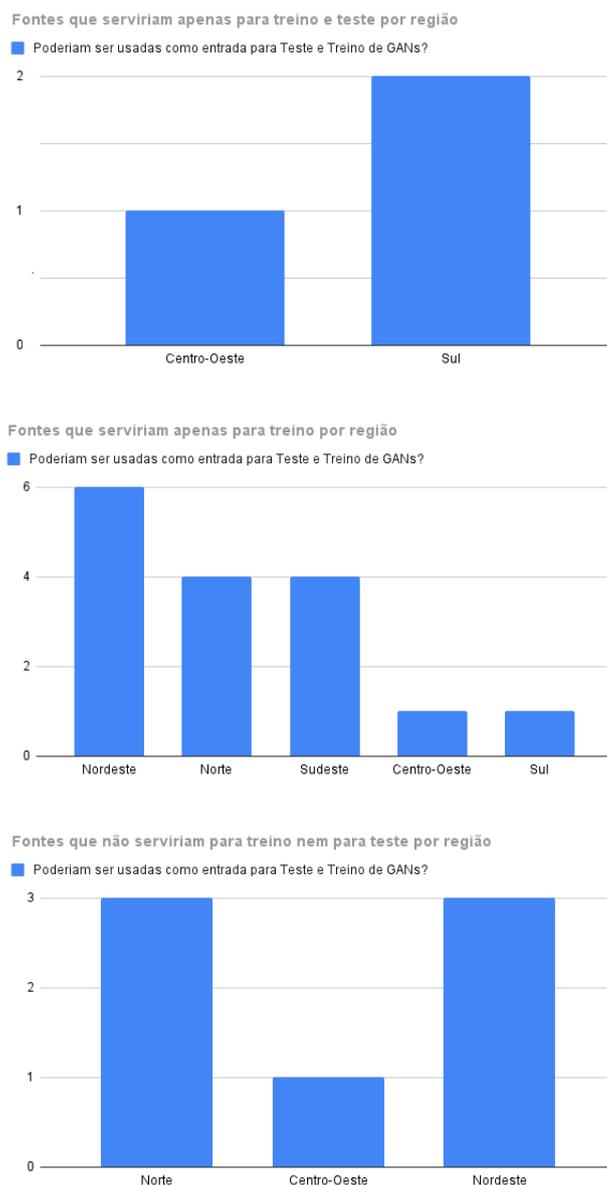
e do contexto de seu desaparecimento. O estado que recebeu essa nota máxima foi o Distrito Federal.

Tabela 5.1: Análise das informações disponíveis sobre pessoas desaparecidas de cada estado brasileiro

Fonte	Dados aparentemente inconsistentes	Imagens de progressão de idade	Nota sobre informações do desaparecimento	Poderiam ser usadas como entrada para Teste e Treino de GANs?
Registro Nacional de Pessoas Desaparecidas	Não se aplica	Não se aplica	0	Não
SINESP Cidadão	Sim	Não se aplica	2	Não
Acre	Sim	Não se aplica	1	Não
Alagoas	Sim, disponível apenas em redes sociais	Não	3	Apenas treino
Amapá	Sim	Não se aplica	1	Não
Amazonas	Não	Não	3	Apenas treino
Bahia	Não	Não	3	Apenas treino
Ceará	Sim, disponível apenas em redes sociais	Não	3	Apenas treino
Distrito Federal	Não	Todas as crianças e adolescentes	5	Treino e Teste
Espirito Santo	Não	Não	3	Apenas treino
Goiás	Não	Não	3	Apenas treino
Maranhão	Não	Não	3	Apenas treino
Mato Grosso	Sim, muitas pessoas sem foto	Não	1	Apenas treino
Mato Grosso do Sul	Sim, disponível apenas em redes sociais	Não	3	Apenas treino
Minas Gerais	Não	Não	3	Apenas treino
Pará	Sim, sem dados divulgados para adultos	Não	3	Apenas treino
Paraíba	Sim	Não se aplica	1	Não
Paraná	Não	Algumas crianças e adolescentes	4	Treino e Teste
Pernambuco	Não	Não	3	Apenas treino
Piauí	Sim	Não se aplica	1	Não
Rio de Janeiro	Não	Não	3	Apenas treino
Rio Grande do Norte	Sim	Não se aplica	1	Não
Rio Grande do Sul	Sim, muitas pessoas sem foto	Não	3	Apenas treino
Rondônia	Sim	Não se aplica	1	Não
Roraima	Não	Não	3	Apenas treino
Santa Catarina	Não	Algumas crianças e adolescentes	4	Treino e Teste
São Paulo	Não	Não	3	Apenas treino
Sergipe	Não	Não	3	Apenas treino
Tocantins	Sim, disponível apenas em redes sociais	Não	3	Apenas treino

Comparativamente, assim como demonstra na Tabela 5.1, os gráficos da Figura 5.1 representam quantos estados podem servir para treino e teste de redes neurais. Para treino e teste, são poucos estados, no Centro-Oeste seria apenas o Distrito Federal e no Sul apenas dois, Paraná e Santa Catarina. Fontes que serviriam para apenas treino, a maioria dos estados da região Nordeste, logo Norte e Sudeste, Centro-Oeste e Sul. Excluindo as ferramentas Registro Nacional de Pessoas Desaparecidas e SINESP, onde deveriam conter todos os dados dos estados, fontes que não podem ser trabalhadas nesta proposta de trabalho são três do Norte, uma do Centro-Oeste e três do Nordeste.

Figura 5.1: Contagem imagens por estados



Fonte: Elaboração própria

Resumidamente, 75% dos dados da região Centro-Oeste pode ser utilizada para treino e 25% para treino e teste. Para a região Nordeste, aproximadamente 66,67% pode ser utilizada para treino, nenhuma

fonte para treino e teste e 33,33% não serviria para as simulações deste trabalho. Região Norte seriam 57,14% para apenas treino e 42,86% são fontes não utilizáveis. Já para a região Sudeste, todos os dados de todos os estados podem ser utilizados apenas para treino. Por último, para a região Sul, 33,33% apenas para treino, 66,67% podem ser utilizadas para treino e teste. Curiosamente a região Sul é onde há mais dados publicados, apesar do estado de São Paulo ser o estado com mais dados de desaparecimento do Brasil como disponível no Anuário Brasileiro de Segurança Pública (Pública 2022). Sete estados não possuem seus dados divulgados corretamente, dezesseis estados não disponibilizam imagens de progressão de idade quando em casos de desaparecimentos de crianças e adolescentes que passaram três anos ou mais desaparecidos e nem de adultos dos quais estão desaparecidos por mais de cinco anos. Apenas o Distrito Federal disponibiliza imagens de progressão de idade para todas as vítimas que deveriam ter.

Tais resultados demonstram a urgência da implementação de um sistema nacional e unificado de todos os estados.

5.2 BANCO DE DADOS

Utilizando ambos os métodos citados no Capítulo 4, manual e automático via código *Python*, foi possível obter um banco de dados com 262 imagens e 1656 imagens respectivamente.

Para quantificar os resultados do método de construção manual do banco, foi criado um sistema de notas para essas imagens. Primeiramente, é extremamente importante coletar boas imagens dessas pessoas desaparecidas. Observando outros trabalhos com GANs, pontos importantes como qualidade de imagem e pose foram considerados neste trabalho para avaliar o quão boa e utilizável uma imagem poderia ser. A Tabela 5.2 é dividida por estado brasileiro e apresenta imagens coletadas, idade do desaparecimento, se existe progressão de idade e qualidade da imagem original e da imagem progredida. Ou seja, essa Tabela foi feita com a média das notas de cada imagem coletada de cada estado. As tabelas inteiras podem ser visualizadas nos Apêndices 1, 2, 3 e 4.

A escala criada para avaliar as imagens foi de notas de 0,5 a 3, por possuir menos critérios de avaliação, essa escala foi diferente da classificação das fontes. Onde 3 é considerada uma imagem perfeita pois é do rosto em um ângulo frontal e alta qualidade, 2 é uma imagem de boa qualidade mas com um postura diferente. A próxima nota é a 1, onde é uma imagem do rosto de ângulo frontal, mas com qualidade muito baixa. A última nota é 0,5, é de uma pose não tão clara, por exemplo, onde o rosto está coberto ou mostra apenas um lado do rosto e de baixa qualidade. A nota zero não foi incluída nessa escala pois assumir uma nota zero seria como se a imagem não servisse de maneira alguma, porém, a própria Polícia já tem esse cuidado de selecionar fotos com uma qualidade e posição boa. Para a coluna de informação, foi utilizada uma escala de 0 a 5, onde nota 5 é dada quando os dados mostram a fotografia original, uma progressão imagem, informações sobre a pessoa e o desaparecimento, idade, dados, local/lugar e informações adicionais, como roupas ou problemas de saúde. A nota 4 é dada aos dados que mostram a fotografia original, uma imagem progredida, idade, dados e local/lugar. A nota 3 a fonte disponibiliza uma fotografia original, idade, dados e local/lugar do desaparecimento. Nota 2 é para quando apenas idade e dados sobre o desaparecimento estão divulgados, mas sem fotografias. Nota 1 é quando existem os dados

no sistema mas não há acesso ou as informações não são claras o suficiente. A última nota 0 é reservado para bancos de dados que não existem ou não fornecem qualquer informação se os dados existem ou não.

Tabela 5.2: Análise das informações disponíveis sobre pessoas desaparecidas de cada estado brasileiro

Fonte	Número de imagens coletadas	Dados aparentemente inconsistentes	Imagens de progressão de idade	Média de qualidade das imagens originais	Média de qualidade das imagens de progressão de idade	Média informação sobre o desaparecimento	Pode ser utilizado para teste e/ou treino?
Registro Nacional de Pessoas Desaparecidas	NA	NA	NA	NA	NA	0	No
SINESP Cidadão	Nenhuma	Sim	NA	NA	NA	2	Não
Bahia	88	Não	NA	2	NA	3	Treino
Distrito Federal	34	Não	Todas as vítimas	2	3	5	Treino e Teste
Rio de Janeiro	58	Não	NA	2	NA	3	Treino
Paraná	82	Não	Algumas crianças e adolescentes	2	2	4	Treino e Teste

A Tabela 5.2 demonstra a importância da qualidade da imagem na hora da coleta e a divulgação dos dados sobre o desaparecimento. A Bahia possui muitas imagens divulgadas, mas não possui progressão de imagem das pessoas desaparecidas e não obteve uma nota tão boa em relação a informação dos desaparecimentos. Em comparação, o Distrito Federal disponibiliza boas fotos, boas imagens de progressão de idade e contextualiza o desaparecimento.

Com as imagens coletadas, foi possível obter um banco de dados com rostos brasileiros. Adicionalmente, dentro do processo de coleta manual, foi possível catalogar diversas características a respeito das imagens o que pode auxiliar em trabalhos futuros para fins estatísticos. Mais uma vez, comparando as imagens disponíveis e os dados do desaparecimento, demonstra o quanto necessário é uma implementação de uma política nacional. Desta maneira, seria possível uma padronização de imagens e dados disponíveis, logo, servindo para aplicação em massa de tecnologias que envolvem *Deep Learning* para progressão de idade. Consequentemente, podendo melhorar a identificação de vítimas.

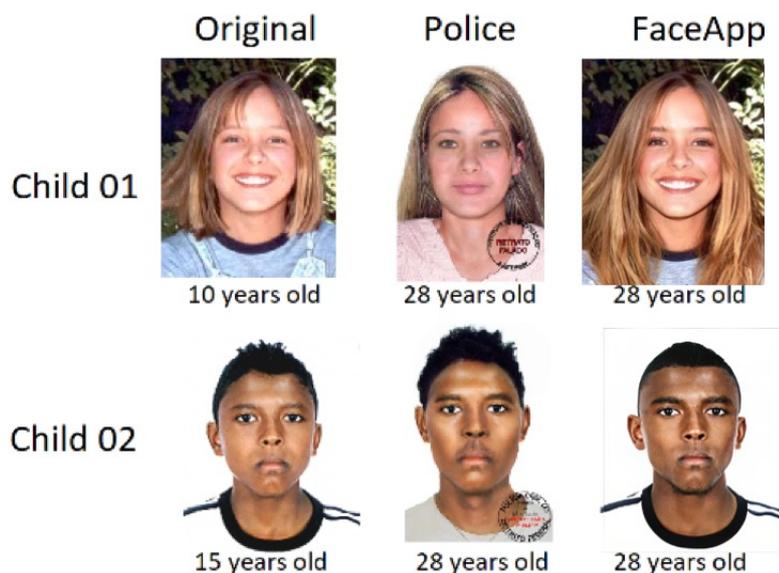
5.3 COMPARAÇÃO COM CASOS REAIS

Devido às limitações do projeto da AttentionGAN, não foi possível realizar o treino até o final e obter resultados por conta da falta do vetor de atributos para o banco construído manualmente e para o banco obtido via automação em *Python*. Além de limitação computacional ocasionada pela ferramenta de execução no caso da implementação do projeto CycleGAN. Contudo, foi possível obter resultados com ferramentas prontas que utilizam Redes Neurais, *FaceApp* e *Face++*. A Figura 5.2 compara a progressão de idade utilizando a imagem feita por métodos tradicionais feitos pela Polícia Civil do Distrito Federal com imagens geradas por redes neurais usando *FaceApp*.

As crianças 01 e 02 seguem desaparecidas. A criança 03 não faz parte do banco de dados da polícia, é usada apenas como exemplo. A criança 03 é a única que possui uma imagem real mais velha. A criança 01 desapareceu aos 10 anos de idade e hoje em dia ela deveria ter 28 anos. Sua progressão de idade é de boa qualidade, mas a imagem no *FaceApp* preservou atributos únicos de seu sorriso. Algumas características de seus dentes e gengivas quando criança foram destacados e preservados, como os sulcos nasolabiais e bolsas sob os olhos. Além disso, no *FaceApp* é possível adicionar um sorriso aberto ou apenas um sorriso suave, diferentes estilos de cabelo, o que traz uma gama de opções e possibilidades que poderia aumentar

sua identificação após anos de desaparecimento. Na Figura 5.2, o adolescente desapareceu quando tinha 15 anos e hoje em dia teria 28 anos de idade. Observando as diferenças entre a imagem da polícia e o resultado da manipulação do *FaceApp*, é notável o aumento de resolução da foto feita pelo aplicativo. O *FaceApp* manteve seus traços como formato da cabeça e nariz, mas adicionando outras características como bolsas nos olhos, dobras nasolabiais, sobrancelhas mais espessas e ressaltou pelos faciais, coisas mais típicas do envelhecimento. A primeira imagem da terceira criança, um exemplo utilizado neste trabalho, é de quando ele tinha 11 anos. A segunda imagem é como esta terceira criança está atualmente com 21 anos de idade. A terceira imagem é editada pelo *FaceApp* nas conformidades da lei, ou seja, se ele tivesse sumido tendo 11 anos, a próxima progressão seria após três anos. A terceira e quarta imagem é a progressão de idade usando *FaceApp* com base na primeira imagem. Já que as imagens são de um adolescente e um jovem adulto, foram adicionadas sobrancelhas espessas, bolsas e linhas de expressão, mandíbula e pomo de Adão destacados. Adicionalmente, a marca da boca foi conservada e o nariz mais grosso. A segunda imagem aparenta precisa em relação à quarta imagem. Pois na quarta imagem, onde a criança já estaria com 21 anos, as orelhas foram conservadas, o sinal da boca, cabelo, as sobrancelhas mais espessas e mandíbula se encaixam em suas características atuais.

Figura 5.2: Comparação entre as progressões de imagem da Polícia e imagens do *FaceApp*.



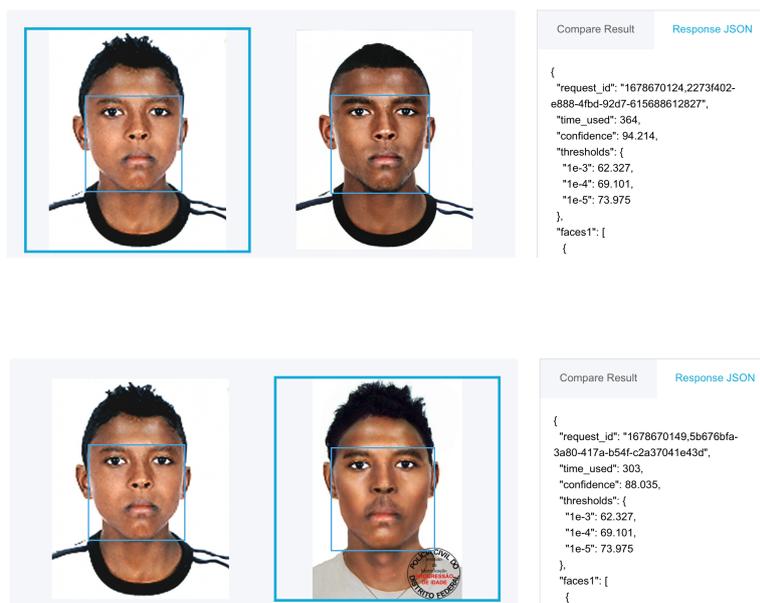


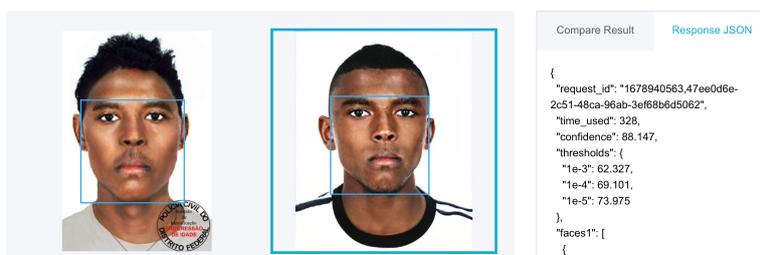
Fonte: Elaboração própria

Utilizando o *Face++* para as três crianças, comparando sua imagem original com a progressão de imagem da Polícia e com as editadas pelo *FaceApp* foi observada uma acurácia maior para as imagens editadas por redes neurais.

Para a criança 1, a acurácia comparativa entre as imagens feita pelo *FaceApp* foi de um valor 94,214. Já testando a similaridade entre progressão de idade da polícia e do *FaceApp* foi de 88,147.

Figura 5.3: Criança 1: Comparação entre a progressão de imagem da Polícia Civil do Distrito Federal e o aplicativo *FaceApp*

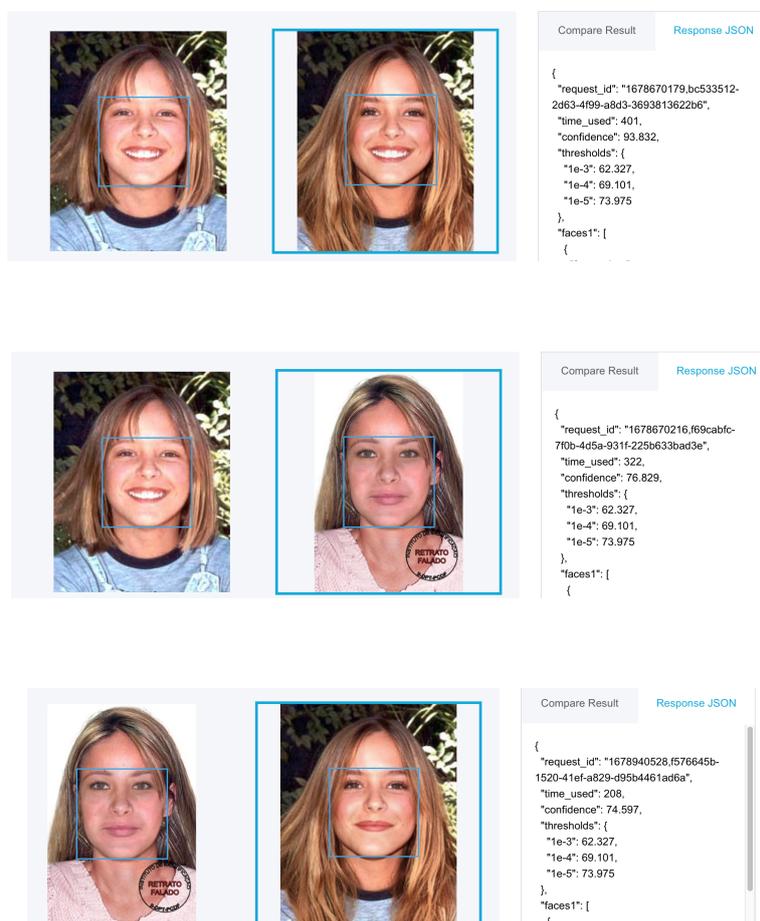




Fonte: Elaboração própria

Para a criança 2, a nota de acurácia foi de 93,832 para a imagem do *FaceApp*. Entre a progressão de idade da polícia e do *FaceApp* foi de 74,597. Foi colocado em sorriso cerrado no *FaceApp* para a criança 2.

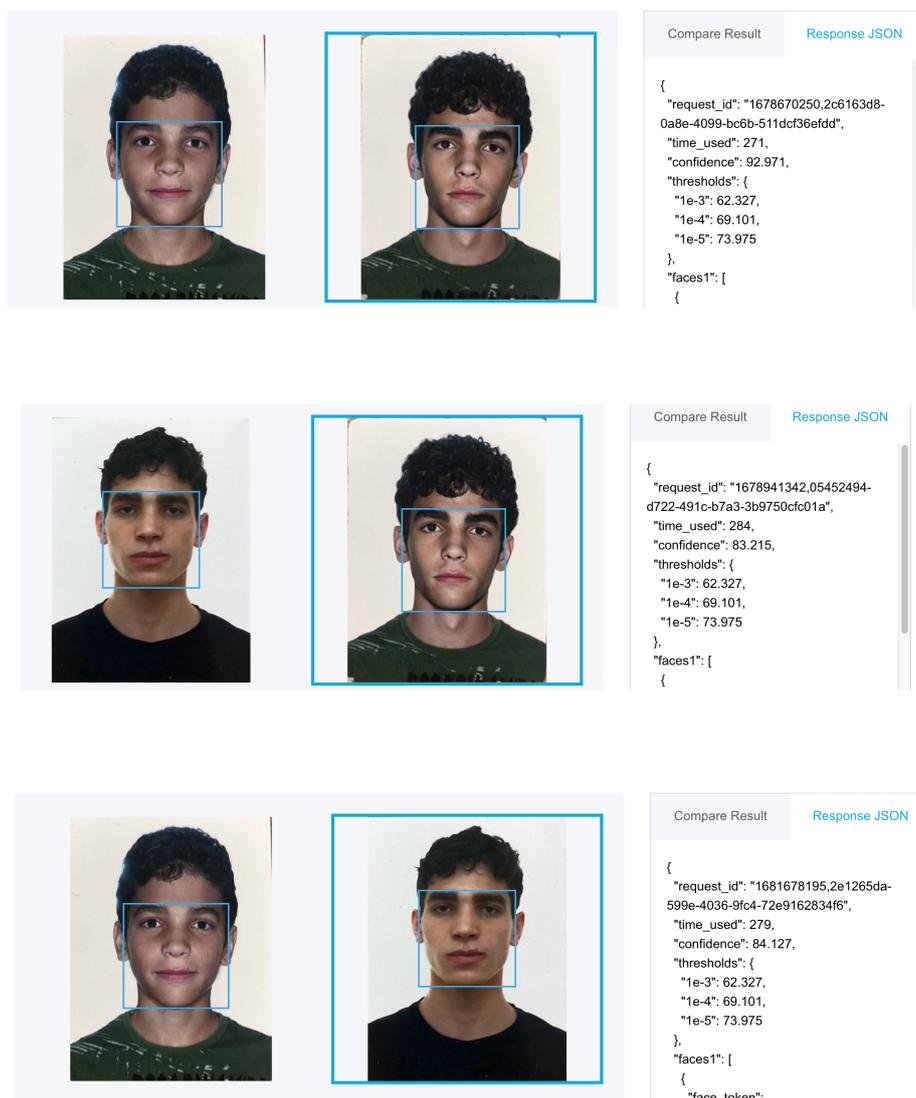
Figura 5.4: Criança 2: Comparação entre a progressão de imagem da Polícia Civil do Distrito Federal e o aplicativo *FaceApp*



Fonte: Elaboração própria

Por fim, para a criança 3, a acurácia obtida foi de 92,971 de similaridade entre a imagem original quando criança e a feita no aplicativo. Já entre a imagem original com 21 anos e a do *FaceApp* foi de 83,215.

Figura 5.5: Criança 3: Comparação entre a progressão de imagem original da criança e do aplicativo *FaceApp*. Logo, a comparação da imagem original da criança com 11 anos e novamente com a atual com 21 anos.



Fonte: Elaboração própria

O objetivo de traçar essa comparação vai de acordo com a contribuição proposta neste trabalho, demonstrar que as Redes Neurais conseguem gerar resultados similares aos de uma progressão de idade feita manualmente. Porém, é válido ressaltar que é uma comparação objetiva, não necessariamente a imagem gerada pelo *FaceApp* valida a aparência de como as crianças do exemplo estariam atualmente.

Os números de acurácia não são para estabelecer qual técnica é melhor, e sim demonstrar o quão similar a criança está de sua progressão de idade. Ao comparar a imagem progredida pela polícia e a imagem progredida pelo *FaceApp*, resulta em uma probabilidade muito alta de serem a mesma pessoa. Em outras palavras, com um banco de dados e as ferramentas corretas, GANs e outros tipos de redes neurais podem ser aliadas nos processos de identificação e progressão de idade dos órgãos de polícia. Uma vez que é possível observar que imagens feitas pelo *FaceApp*. Em órgãos que não possuem nenhuma progressão de idade feita e se faz necessário realizar de muitas pessoas, é possível utilizar esse artifício das GANs para

essa tarefa. Infelizmente, o *FaceApp* não faz a edição em lote, mas ainda se mostrou útil em questão de envelhecimento e preservação das características faciais.

6 CONCLUSÃO

No Brasil, os números de desaparecimentos ainda são alarmantes (Pública 2022). Embora haja um esforço contínuo das forças de segurança pública, ainda falta uma modernização e interoperabilidade entre os órgãos Brasileiros. Apesar de estabelecida a Lei 13.812/2019 (República), ainda não existe um banco de dados unificado e uma padronização de informações. Muitas forças policiais no Brasil não divulgam seus dados e nem realizam imagens de progressão de idade para vítimas que deveriam ter. Tais fatos dificultam a identificação de pessoas desaparecidas. Devido ao alto volume de imagens e informações, a técnica de aprendizado de máquina e redes neurais se faz útil na solução de algumas questões ainda presentes nesse cenário Brasileiro.

Unificar os dados, padronizar as formas de divulgação e cumprir com as progressões de idade podem ser soluções que auxiliem na identificação de vítimas. Visto que ainda é um problema no Brasil, este trabalho tomou como motivação para demonstrar o quão boas as informações estão, como são disponibilizadas e como *Deep Learning* pode auxiliar os órgãos de segurança pública de cada estado. Para cumprir estes objetivos, neste trabalho, foram estabelecidos dois passos. O primeiro deles foi a pesquisa, coleta de dados e extração de informações sobre estes dados, de forma manual e automática. O segundo passo foi utilizar tais dados coletados para simulações utilizando redes neurais. Seguindo esses passos, por fim, foi possível obter resultados significantes que auxiliem no cumprimento da Lei 13.812/2019 (República) e também que somam o processo de progressão de idade nas Polícias Civis.

Cumprindo os objetivos e seguindo a metodologia imposta, os resultados foram promissores. Foi possível mapear as diversas fontes oficiais do país, além de catalogá-las, foi feita uma classificação de como a divulgação das vítimas de desaparecimento é feita, como acessar essas informações e sua usabilidade para futuras automações. Com as fontes mapeadas e classificadas em quais poderiam servir para um treinamento e teste de uma rede neural, foi criado um banco de dados com essas imagens Brasileiras e suas respectivas progressão de idade quando cabível. A respeito do banco de imagens, para as imagens extraídas manualmente, foram feitas planilhas onde descreve cada informação, por exemplo, idade do desaparecimento, nome da imagem, se existe ou não progressão e outras informações que podem ser utilizadas para gerar estatísticas em trabalhos futuros. Adicionalmente, com o banco de imagens construído, foi possível testar três abordagens utilizando redes neurais. A primeira delas, *AttentionGAN*, o vetor de atributos foi um impeditivo para completar o treino, teste e validação com as imagens brasileiras. Ao testar a *AttentionGAN* com o banco utilizado no próprio projeto, foi possível observar que de fato a rede GAN faz a tradução de imagem entre domínios. Segundo, foi testado o projeto *CycleGAN* e os recursos computacionais foram impeditivos nesta etapa, não foi possível obter imagens claras com tão poucas épocas de execução. Por fim, algumas imagens foram testadas em aplicações que não necessitam de programação, apenas a manipulação via interface gráfica do aplicativo. Utilizando *FaceApp* e *Face++*, ambas redes neurais, nessa etapa, os resultados foram favoráveis pois demonstraram uma conservação das características fenotípicas das pessoas e ainda conservando uma alta similaridade com as imagens originais.

Os resultados do mapeamento das fontes e a criação do banco de dados com imagens brasileiras demonstram o quão se faz necessária essa ação por parte dos órgãos de segurança pública e o quão benéfico

seria se houvesse essa integração e interoperabilidade. Uma vez que, havendo o banco unificado e uma integração entre os órgãos, seria possível aplicar tecnologias de aprendizado de máquina que extraíam dados estatísticos e também auxiliar os profissionais papiloscopistas nas criações das imagens de progressão de idade. Especialmente para aquelas vítimas que não possuem seus dados devidamente divulgados, não possui imagens originais, ou não possui sua imagem progredida mesmo após anos depois do seu desaparecimento.

Como trabalhos futuros para continuar o desenvolvimento deste trabalho, seria interessante a publicação do banco de dados para novas simulações com as GANs. Além da criação do vetor de atributos de cada imagem coletada, desta forma seria possível completar o projeto *AttentionGAN* com imagens brasileiras e obter uma rede neural treinada com exemplos brasileiros.

REFERÊNCIAS BIBLIOGRÁFICAS

Abadi et al. 2016 ABADI, M.; BARHAM, P.; CHEN, J.; CHEN, Z.; DAVIS, A.; DEAN, J.; DEVIN, M.; GHEMAWAT, S.; IRVING, G.; ISARD, M. et al. Tensorflow: A system for large-scale machine learning. In: *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*. [S.l.: s.n.], 2016. p. 265–283.

Acayaba ACAYABA, C. Brasil registra média de 200 desaparecidos por dia, diz anuário brasileiro de segurança pública. *Portal G1 - Globo*. Disponível em: <<https://g1.globo.com/sp/sao-paulo/noticia/2022/06/28/brasil-registra-media-de-200-desaparecidos-por-dia-diz-anuario-brasileiro-de-seguranca-publica.ghtml>>.

Alencar et al. 2022 ALENCAR, C. L. F. d.; SANTIAGO, A. P. d. A. d. C. e. S.; SOUZA, C. A. d.; MELO, C. V. M. d.; LIMA, J. J. B. d.; SOUSA, V. L. V. R.; MOREIRA, M. H. B. d. A.; SILVA, J. A. C. d.; RIBEIRO, T. B.; BIONE, E. C.; XIMENES, W. d. S.; PEREIRA, M. E. K. C.; DAMACENA, W. S. Missing persons in brazil: use of the genetic profiles bank and search strategies. *Research, Society and Development*, v. 11, n. 8, p. e16011830305, Jun. 2022. Disponível em: <<https://rsdjournal.org/index.php/rsd/article/view/30305>>.

Amazonas AMAZONAS, P. C. do Estado do. *Desaparecidos*. <<http://www.policiacivil.am.gov.br/desaparecidos/>>.

Anaconda Software Distribution 2020 ANACONDA Software Distribution. Anaconda Inc., 2020. Disponível em: <<https://docs.anaconda.com/>>.

Atienza 2018 ATIENZA, R. *Advanced Deep Learning with Keras: Apply Deep Learning Techniques, Autoencoders, GANs, Variational Autoencoders, Deep Reinforcement Learning, Policy Gradients, and More*. [S.l.]: Packt Publishing, 2018. ISBN 1788629418.

Bisong 2019 BISONG, E. Google colaboratory. In: _____. *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*. Berkeley, CA: Apress, 2019. p. 59–64. ISBN 978-1-4842-4470-8. Disponível em: <https://doi.org/10.1007/978-1-4842-4470-8_7>.

Bradski 2000 BRADSKI, G. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

Bragias, Hine e Fleet 2021 BRAGIAS, A.; HINE, K.; FLEET, R. ‘only in our best interest, right?’ public perceptions of police use of facial recognition technology. *Police Practice and Research*, v. 22, p. 1–18, 06 2021.

Catarina CATARINA, P. C. E. de S. *Desaparecidos*. <<https://desaparecidos.pc.sc.gov.br/#/>>.

Catarina CATARINA, P. M. E. de S. *SOS Desaparecidos*. <<https://www.pm.sc.gov.br/sos-desaparecidos>>.

Ceará CEARÁ, P. C. do Estado do Ceará 12ª Delegacia do DHPP do. *Desaparecidos - DHPP Ceará*. <<https://www.instagram.com/desaparecidosdhppce/>>.

Chandaliya e Nain 2021 CHANDALIYA, P. K.; NAIN, N. Child face age progression and regression using self-attention multi-scale patch gan. In: *2021 IEEE International Joint Conference on Biometrics (IJCB)*. [S.l.: s.n.], 2021. p. 1–8.

Chandran et al. 2018 CHANDRAN, P. S.; BYJU, N. B.; DEEPAK, R. U.; NISHAKUMARI, K. N.; DEVANAND, P.; SASI, P. M. Missing child identification system using deep learning and multiclass svm. In: *2018 IEEE Recent Advances in Intelligent Computational Systems (RAICS)*. [S.l.: s.n.], 2018. p. 113–116.

Choi et al. 2018 CHOI, Y.; CHOI, M.; KIM, M.; HA, J.-W.; KIM, S.; CHOO, J. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2018.

CNMP CNMP, C. N. do M. P. O que é o sinalid. *Comissão de Defesa dos Direitos Fundamentais*. Disponível em: <<https://www.cnmp.mp.br/portal/institucional/comissoes/comissao-de-defesa-dos-direitos-fundamentais/sinalid/informacoes-sobre-o-sistema>>.

Comunicação/DGPC COMUNICAÇÃO/DGPC, D. de. Desaparecidos / progressão de idade. *Polícia Civil Distrito Federal*. Disponível em: <<https://www.pcdf.df.gov.br/servicos/desaparecidos#>>.

Comunicação/DGPC COMUNICAÇÃO/DGPC, D. de. Progressão de idade auxilia na busca de crianças desaparecidas. *Polícia Civil Distrito Federal*. Disponível em: <<https://www.pcdf.df.gov.br/noticias/8086/progressao-de-idade-auxilia-na-busca-de-criancas-desaparecidas>>.

Developer 2023 DEVELOPER, N. *CUDA Toolkit*. [S.l.], 2023. Disponível em: <<https://developer.nvidia.com/cuda-toolkit>>.

EXPRESSO EXPRESSO. *Consultar pessoas desaparecidas e cadáveres não identificados*. <<https://www.go.gov.br/servicos-digitais/dgpc/consulta-de-pessoas-desaparecidas-e-cadaveres-nao-identificados>>.

Face++ 2017 FACE++. *Face Comparing*. 2017. <<https://console.faceplusplus.com/documents/5679308>>.

Fisher FISHER, M. A revealing map of the world's most and least ethnically diverse countries. *Washington Post*. Disponível em: <<https://www.washingtonpost.com/news/worldviews/wp/2013/05/16/a-revealing-map-of-the-worlds-most-and-least-ethnically-diverse-countries/>>.

Gan GAN, N. Facial recognition helps reunite kidnapped toddler with family after 32 years. *CNN*. Disponível em: <<https://edition.cnn.com/2020/05/19/asia/china-kidnapped-son-reunited-intl-hnk/index.html>>.

Geraiis GERAIS, D. D. R. D. P. D. P. C. E. de M. *ALBUM DE DESAPARECIDOS*. <<https://desaparecidos.policiacivil.mg.gov.br/desaparecido/album/>>.

Goncharov 2017 GONCHAROV, Y. *FaceApp*. 2017. <<https://www.faceapp.com>>.

Goodfellow GOODFELLOW, I. *4.5 years of GAN progress on face generation*. <https://twitter.com/goodfellow_ian/status/1084973596236144640>.

Goodfellow et al. 2014 GOODFELLOW, I. J.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDEFARLEY, D.; OZAIR, S.; COURVILLE, A.; BENGIO, Y. *Generative Adversarial Networks*. arXiv, 2014. Disponível em: <<https://arxiv.org/abs/1406.2661>>.

Grimmer, Raghavendra e Busch 2021 GRIMMER, M.; RAGHAVENDRA, R.; BUSCH, C. Deep face age progression: A survey. *IEEE Access*, PP, p. 1–1, 06 2021.

Grosso GROSSO, P. J. C. E. de M. *Pessoas Desaparecidas*. <<https://desaparecidos.pjc.mt.gov.br/pessoa-desaparecida/5095>>.

Géron 2017 GÉRON, A. *Hands-on machine learning with Scikit-Learn and TensorFlow : concepts, tools, and techniques to build intelligent systems*. Sebastopol, CA: O'Reilly Media, 2017. ISBN 978-1491962299.

Janeiro JANEIRO, P. C. do Estado do Rio de. *Portal de Desaparecidos*. <<https://desaparecidos.pcivil.rj.gov.br>>.

Kamble e Dale 2021 KAMBLE, V.; DALE, M. Face recognition of children using ai classification approaches. In: *2021 International Conference on Emerging Smart Computing and Informatics (ESCI)*. [S.l.: s.n.], 2021. p. 248–251.

Karras et al. 2019 KARRAS, T.; LAINE, S.; AITTALA, M.; HELLSTEN, J.; LEHTINEN, J.; AILA, T. *Analyzing and Improving the Image Quality of StyleGAN*. arXiv, 2019. Disponível em: <<https://arxiv.org/abs/1912.04958>>.

Karras et al. 2020 KARRAS, T.; LAINE, S.; AITTALA, M.; HELLSTEN, J.; LEHTINEN, J.; AILA, T. Analyzing and improving the image quality of StyleGAN. In: *Proc. CVPR*. [S.l.: s.n.], 2020.

Krizhevsky, Sutskever e Hinton 2012 KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: PEREIRA, F.; BURGESS, C.; BOTTOU, L.; WEINBERGER, K. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2012. v. 25. Disponível em: <<https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>>.

Langr e Bok 2019 LANGR, J.; BOK, V. *Gans in action: Deep learning with generative adversarial networks*. [S.l.]: Manning, 2019.

Ledig et al. 2016 LEDIG, C.; THEIS, L.; HUSZAR, F.; CABALLERO, J.; CUNNINGHAM, A.; ACOSTA, A.; AITKEN, A.; TEJANI, A.; TOTZ, J.; WANG, Z.; SHI, W. *Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network*. arXiv, 2016. Disponível em: <<https://arxiv.org/abs/1609.04802>>.

Liu et al. 2015 LIU, Z.; LUO, P.; WANG, X.; TANG, X. Deep learning face attributes in the wild. In: *Proceedings of International Conference on Computer Vision (ICCV)*. [S.l.: s.n.], 2015.

Maranhão MARANHÃO, S. de Segurança Pública do. *DISQUE DENÚNCIA - DESAPARECIDOS*. <<https://www.ssp.ma.gov.br/disque-denuncia/desaparecidos/>>.

Matcha 2019 MATCHA, A. C. N. Implementing cyclegan for age conversion. *PaperSpace Blog*, 2019. Disponível em: <<https://blog.paperspace.com/use-cyclegan-age-conversion-keras-python/>>.

Merkel 2014 MERKEL, D. Docker: lightweight linux containers for consistent development and deployment. *Linux journal*, v. 2014, n. 239, p. 2, 2014.

Moraes et al. 2022 MORAES, J. K. de; BALDASSO, R. P.; GONÇALVES, P.; BEUX, L.; FERNANDES, M. M.; STRAPASSON, R. A. P. Two-dimensional facial approximation: facial composite and digital 3d facial approximation techniques applied on cone-beam tomography. *Conjecturas*, v. 22, n. 7, p. 183–199, jul. 2022. Disponível em: <<https://conjecturas.org/index.php/edicoes/article/view/1189>>.

Muthukadan 2018 MUTHUKADAN, B. Selenium with python. 2018. Disponível em: <<https://selenium-python.readthedocs.io/navigating.html>>.

Norte NORTE, P. C. E. do Rio Grande do. *Desaparecidos*. <<http://www.policiacivil.rn.gov.br/Conteudo.asp?TRAN=ITEM&TARG=275629&ACT=&PAGE=0&PARM=&LBL=Desaparecidos>>.

Paraná PARANÁ, P. C. do Estado do. *Pessoas desaparecidas*. <<https://www.desaparecidos.pr.gov.br/desaparecidos/>>.

Pará PARÁ, D. P. C. E. do. *Crianças e adolescentes desaparecidos*. <<http://data.policiaivil.pa.gov.br>>.

Paszke et al. 2019 PASZKE, A.; GROSS, S.; MASSA, F.; LERER, A.; BRADBURY, J.; CHANAN, G.; KILLEEN, T.; LIN, Z.; GIMELSHEIN, N.; ANTIGA, L.; DESMAISON, A.; KOPF, A.; YANG, E.; DEVITO, Z.; RAISON, M.; TEJANI, A.; CHILAMKURTHY, S.; STEINER, B.; FANG, L.; BAI, J.; CHINTALA, S. Pytorch: An imperative style, high-performance deep learning library. In: *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019. p. 8024–8035. Disponível em: <<http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>>.

Paulo PAULO, P. C. do Estado de S. *Institucional | Retrato Falado*. <<https://www.ssp.sp.gov.br/fale/institucional/answers.aspx?t=12>>.

Paulo PAULO, S. de Segurança Pública do Estado de S. *Pessoas Desaparecidas*. <https://www.ssp.sp.gov.br/servicos/pessoas_desaparecidas.aspx>.

Pernambuco PERNAMBUCO, P. C. E. do. *ADULTOS DESAPARECIDOS*. <<http://www.policiaivil.pe.gov.br/adultos-desaparecidos>>.

Pernambuco PERNAMBUCO, P. C. E. do. *CRIANÇAS DESAPARECIDAS*. <<http://www.policiaivil.pe.gov.br/criancas-desaparecidas>>.

Pessoa PESSOA, P. C. do Estado da Bahia Delegacia de P. *DESAPARECIDOS*. <<https://disquedenuncia.com/desaparecidos/>>.

PLID/AL PLID/AL. *PLID/AL-DESAPARECIDOS ALAGOAS*. <<https://www.instagram.com/sinalidplidal/>>.

Pública 2022 PÚBLICA, F. B. de S. Anuário brasileiro de segurança pública. p. 92–100, 2022.

Pública 2020 PÚBLICA, M. da Justiça e S. *Sinesp Cidadão*. 2020. <<https://www.gov.br/pt-br/apps/sinesp-cidadao>>.

Público PÚBLICO, S. de Comunicação Social Conselho Nacional do M. Cnmp assina acordo para criação do cadastro nacional de desaparecidos. *Conselho Nacional do Ministério Público*. Disponível em: <<https://www.cnmp.mp.br/portal/todas-as-noticias/15270-cnmp-assina-acordo-de-cooperacao-para-implementar-o-cadastro-nacional-de-pessoas-desaparecidas-2>>.

rajkumar R. Karthick Raja RAJKUMAR R. KARTHICK RAJA, S. S. G. V. R. S. M. A. D. Identification of missing person in the crowd using pretrained neural network. In: . [s.n.]. Disponível em: <<https://www.irjet.net/archives/V7/i2/IRJET-V7I285.pdf>>.

REIS REIS, P. R. B. Confiabilidade do retrato falado construído com base em relatos de vítimas e/ou de testemunhas de crime como meio de identificação de autoria delituosa. *Conteúdo Jurídico*.

República REPÚBLICA, S.-G. e. S. p. A. J. Presidência da. Lei nº 12.527, de 18 de novembro de 2011. *Diário Oficial da União*. Disponível em: <https://www.planalto.gov.br/ccivil_03/_ato2011-2014/2011/lei/112527.htm>.

República REPÚBLICA, S.-G. e. S. p. A. J. Presidência da. Lei nº 13.812, de 16 de março de 2019. *Diário Oficial da União*. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2019-2022/2019/lei/113812.htm>.

Richardson 2007 RICHARDSON, L. Beautiful soup documentation. *April*, 2007. Disponível em: <<https://beautiful-soup-4.readthedocs.io>>.

Roraima RORAIMA, P. C. E. do. *Desaparecidos*. <<http://www.pc.rr.gov.br/servicos/desaparecidos>>.

ROY e ARUNACHALAM ROY, L.; ARUNACHALAM, S. *AI and machine learning adoption is the future of security and law enforcement*. Disponível em: <<https://www.forbesindia.com/article/isbinsight/ai-and-machine-learning-adoption-is-the-future-of-security-and-law-enforcement/72807/1>>.

Rumelhart, Hinton e Williams 1986 RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning Representations by Back-propagating Errors. *Nature*, v. 323, n. 6088, p. 533–536, 1986. Disponível em: <<http://www.nature.com/articles/323533a0>>.

Sergipe SERGIPE, P. C. E. de. *Desaparecidos*. <<https://policiacivil.se.gov.br/desaparecidos/>>.

(SESP) (SESP), S. de Estado da Segurança Pública e Defesa Social do E. S. *Portal de Pessoas Desaparecidas*. <<https://disquedenuncia181.es.gov.br/desaparecidos>>.

Sewak, Karim e Pujari 2018 SEWAK, M.; KARIM, M.; PUJARI, P. *Practical Convolutional Neural Networks: Implement advanced deep learning models using Python*. Packt Publishing, 2018. ISBN 9781788394147. Disponível em: <<https://books.google.com.br/books?id=bOIODwAAQBAJ>>.

Sharma, Sharma e Jindal 2021 SHARMA, N.; SHARMA, R.; JINDAL, N. Prediction of face age progression with generative adversarial networks. *Multimedia Tools and Applications*, v. 80, p. 33911 – 33935, 2021.

Sul SUL, D. O. do Estado de Mato Grosso do. *Pessoas Desaparecidas*. <<http://devir.pc.ms.gov.br/#/>>.

Sul SUL, P. C. E. de M. G. *Delegacia de Homicídios MS - SETOR DE PESSOAS DESAPARECIDAS*. <<https://www.instagram.com/desaparecidos.deh.pcms/>>.

Sul SUL, P. C. E. do Rio Grande do. *Desaparecidos*. <<https://www.pc.rs.gov.br/desaparecidos>>.

Tang et al. 2021 TANG, H.; LIU, H.; XU, D.; TORR, P. H.; SEBE, N. Attentiongan: Unpaired image-to-image translation using attention-guided generative adversarial networks. *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 2021.

Tang et al. 2019 TANG, H.; XU, D.; SEBE, N.; YAN, Y. Attention-guided generative adversarial networks for unsupervised image-to-image translation. In: *International Joint Conference on Neural Networks (IJCNN)*. [S.l.: s.n.], 2019.

TO TO, P. *Delegacia de Polícia Interestadual e Captura*. <<https://www.facebook.com/polinterTO/>>.

Ubuntu releases 2018 Ubuntu releases. *Ubuntu 18.04.6 LTS (Bionic Beaver)*. [S.l.], 2018. Disponível em: <<https://releases.ubuntu.com/18.04/>>.

Zhang Zhifei e Qi 2017 ZHANG ZHIFEI, S. Y.; QI, H. Age progression/regression by conditional adversarial autoencoder. In: IEEE. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.], 2017.

Zhu et al. 2017 ZHU, J.-Y.; PARK, T.; ISOLA, P.; EFROS, A. A. *Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks*. arXiv, 2017. Disponível em: <<https://arxiv.org/abs/1703.10593>>.

APÊNDICES

I.1 TABELAS DE CLASSIFICAÇÃO POR IMAGEM POR CADA ESTADO BRASILEIRO EXTRAÍDO MANUALMENTE

Tabela 1: Classificação das imagens do estado da Bahia

Idade do desaparecimento	Nome da imagem	Existe Progressão de idade?	Qualidade da imagem original	Qualidade da imagem de progressão
18	001A18.webp	Não	3	Não se aplica
39	002A39.webp	Não	2	Não se aplica
19	003A19.webp	Não	3	Não se aplica
22	004A22.webp	Não	0,5	Não se aplica
31	005A31.webp	Não	2	Não se aplica
45	006A45.webp	Não	3	Não se aplica
68	007A68.webp	Não	0,5	Não se aplica
48	008A48.webp	Não	1	Não se aplica
08	009A08.webp	Não	1	Não se aplica
46	010A46.webp	Não	3	Não se aplica
33	011A33.webp	Não	0,5	Não se aplica
34	012A34.webp	Não	0,5	Não se aplica
55	013A55.webp	Não	1	Não se aplica
60	014A60.webp	Não	3	Não se aplica
32	015A32.webp	Não	3	Não se aplica
36	016A36.webp	Não	1	Não se aplica
22	017A22.webp	Não	2	Não se aplica
48	018A48.webp	Não	1	Não se aplica
12	019A12.webp	Não	1	Não se aplica
15	020A15.webp	Não	2	Não se aplica
31	021A31.webp	Não	2	Não se aplica
25	022A25.webp	Não	2	Não se aplica
38	023A38.webp	Não	1	Não se aplica
35	024A35.webp	Não	2	Não se aplica
18	025A18.webp	Não	3	Não se aplica
22	026A22.webp	Não	3	Não se aplica
13	027A13.webp	Não	3	Não se aplica
21	027A21.webp	Não	2	Não se aplica
24	028A24.webp	Não	3	Não se aplica

13	029A13.webp	Não	0,5	Não se aplica
39	030A39.webp	Não	1	Não se aplica
42	031A42.webp	Não	1	Não se aplica
49	032A49.webp	Não	3	Não se aplica
12	033A12.webp	Não	3	Não se aplica
23	034A23.webp	Não	2	Não se aplica
79	035A79.webp	Não	3	Não se aplica
18	036A18.webp	Não	3	Não se aplica
18	037A18.webp	Não	3	Não se aplica
53	038A53.webp	Não	3	Não se aplica
26	039A26.webp	Não	1	Não se aplica
23	040A23.webp	Não	3	Não se aplica
20	041A20.webp	Não	3	Não se aplica
59	042A59.webp	Não	2	Não se aplica
58	043A58.webp	Não	1	Não se aplica
14	044A14.webp	Não	2	Não se aplica
51	045A51.webp	Não	3	Não se aplica
18	046A18.webp	Não	3	Não se aplica
35	047A35.webp	Não	3	Não se aplica
37	048A37.webp	Não	3	Não se aplica
14	049A14.webp	Não	2	Não se aplica
16	050A16.webp	Não	0,5	Não se aplica
11	051A11.webp	Não	2	Não se aplica
14	052A14.webp	Não	1	Não se aplica
20	053A20.webp	Não	1	Não se aplica
15	054A15.webp	Não	0,5	Não se aplica
24	055A24.webp	Não	1	Não se aplica
39	056A39.webp	Não	3	Não se aplica
15	057A15.webp	Não	2	Não se aplica
20	058A20.webp	Não	2	Não se aplica
50	059A50.webp	Não	2	Não se aplica
27	060A27.webp	Não	1	Não se aplica
32	061A32.webp	Não	2	Não se aplica
18	062A18.webp	Não	2	Não se aplica
15	063A15.webp	Não	2	Não se aplica
21	064A21.webp	Não	2	Não se aplica
17	065A17.webp	Não	3	Não se aplica
35	066A35.webp	Não	3	Não se aplica
38	067A38.webp	Não	1	Não se aplica
18	068A18.webp	Não	2	Não se aplica

52	069A52.webp	Não	0,5	Não se aplica
11	070A11.webp	Não	2	Não se aplica
19	071A19.webp	Não	3	Não se aplica
52	072A52.webp	Não	2	Não se aplica
30	073A30.webp	Não	3	Não se aplica
13	074A13.webp	Não	1	Não se aplica
12	075A12.webp	Não	2	Não se aplica
45	076A45.webp	Não	2	Não se aplica
14	077A14.webp	Não	2	Não se aplica
22	078A22.webp	Não	0,5	Não se aplica
18	079A18.webp	Não	3	Não se aplica
17	080A17.webp	Não	3	Não se aplica
18	081A18.webp	Não	1	Não se aplica
14	082A14.webp	Não	1	Não se aplica
17	083A17.webp	Não	1	Não se aplica
10	084A10.webp	Não	3	Não se aplica
17	085A17.webp	Não	1	Não se aplica
15	086A15.webp	Não	1	Não se aplica
15	087A15.webp	Não	2	Não se aplica

Tabela 2: Classificação das imagens do estado do Distrito Federal

Idade do desaparecimento	Nome da imagem	Existe Progressão de idade?	Qualidade da imagem original	Qualidade da imagem de progressão
15	001A15.jpg	Sim	3	2
28	001A28P.jpg	Sim	3	2
10	002A10.jpg	Sim	1	3
22	002A22P.jpg	Sim	1	3
13	003A13.jpg	Sim	3	3
27	003A27P.jpg	Sim	3	3
14	004A14.jpg	Sim	2	3
18	004A18P.jpg	Sim	2	3
49	005A49.jpg	Sim	3	1
72	005A72P.jpg	Sim	3	1
15	006A15.jpg	Sim	3	3
29	006A29P.jpg	Sim	3	3
13	007A13.jpg	Sim	1	3
35	007A35P.jpg	Sim	1	3
15	008A15.jpg	Sim	2	3
20	008A20P.jpg	Sim	2	3
10	009A10.jpg	Sim	3	1
28	009A28P.jpg	Sim	3	1
16	010A16.jpg	Sim	1	3
30	010A30P.jpg	Sim	1	3
19	011A19.jpg	Sim	1	3
34	011A34P.jpg	Sim	1	3
16	012A16.jpg	Sim	3	3
31	012A31P.jpg	Sim	3	3
11	013A11.jpg	Sim	1	3
33	013A33P.jpg	Sim	1	3
13	014A13.jpg	Sim	1	3
26	014A26P.jpg	Sim	1	3
08	015A08.jpg	Sim	2	1
28	015A28P.jpg	Sim	2	1
15	016A15.jpg	Sim	1	3
31	016A31P.jpg	Sim	1	3
14	017A14.jpg	Sim	1	1
28	017A28P.jpg	Sim	1	1

Tabela 3: Classificação das imagens do estado do Rio de Janeiro

Idade do desaparecimento	Nome da imagem	Existe Progressão de idade?	Qualidade da imagem original	Qualidade da imagem de progressão
41	001A41.jpg	No	2	Não se aplica
31	002A31.jpg	No	3	Não se aplica
48	003A48.jpg	No	2	Não se aplica
15	004A15.jpg	No	3	Não se aplica
16	005A16.jpg	No	3	Não se aplica
14	006A14.jpg	No	2	Não se aplica
15	007A15.jpg	No	3	Não se aplica
13	008A13.jpg	No	1	Não se aplica
16	009A16.jpg	No	0,5	Não se aplica
17	010A17.jpg	No	0,5	Não se aplica
52	011A52.jpg	No	3	Não se aplica
49	012A49.jpg	No	2	Não se aplica
32	013A32.jpg	No	3	Não se aplica
25	014A25.jpg	No	3	Não se aplica
69	015A69.jpg	No	3	Não se aplica
12	016A12.jpg	No	1	Não se aplica
27	017A27.jpg	No	2	Não se aplica
13	018A13.jpg	No	2	Não se aplica
55	019A55.jpg	No	1	Não se aplica
39	020A39.jpg	No	3	Não se aplica
34	021A34.jpg	No	1	Não se aplica
56	022A56.jpg	No	3	Não se aplica
13	023A13.jpg	No	2	Não se aplica
64	024A64.jpg	No	1	Não se aplica
61	025A61.jpg	No	2	Não se aplica
75	026A75.jpg	No	3	Não se aplica
45	027A45.jpg	No	2	Não se aplica
13	028A13.jpg	No	0,5	Não se aplica
23	029A23.jpg	No	3	Não se aplica
32	030A32.jpg	No	2	Não se aplica
20	031A20.jpg	No	3	Não se aplica
51	032A51.jpg	No	1	Não se aplica
72	033A72.jpg	No	0,5	Não se aplica
19	034A19.jpg	No	0,5	Não se aplica
15	035A15.jpg	No	3	Não se aplica
76	036A76.jpg	No	2	Não se aplica

21	037A21.jpg	No	2	Não se aplica
18	038A18.jpg	No	3	Não se aplica
70	039A70.jpg	No	3	Não se aplica
39	040A39.jpg	No	3	Não se aplica
22	041A22.jpg	No	0,5	Não se aplica
74	042A74.jpg	No	2	Não se aplica
14	043A14.jpg	No	2	Não se aplica
61	044A61.jpg	No	0,5	Não se aplica
08	045A08.jpg	No	2	Não se aplica
33	046A33.jpg	No	2	Não se aplica
55	047A55.jpg	No	2	Não se aplica
27	048A27.jpg	No	3	Não se aplica
58	049A58.jpg	No	3	Não se aplica
64	050A64.jpg	No	2	Não se aplica
33	051A33.jpg	No	2	Não se aplica
21	052A21.jpg	No	3	Não se aplica
26	053A26.jpg	No	2	Não se aplica
28	054A28.jpg	No	3	Não se aplica
43	055A43.jpg	No	0,5	Não se aplica
27	056A27.jpg	No	3	Não se aplica
23	057A23.jpg	No	3	Não se aplica
81	058A81.jpg	No	3	Não se aplica

Tabela 4: Classificação das imagens do estado do Paraná

Idade do desaparecimento	Nome da imagem	Existe Progressão de idade?	Qualidade da imagem original	Qualidade da imagem de progressão
11	001A11.jpg	Não	0,5	Não se aplica
03	002A03.jpg	Não	2	Não se aplica
06	003A06.jpg	Não	2	Não se aplica
11	004A11.jpg	Não	3	Não se aplica
10	005A10.jpg	Não	1	Não se aplica
10	006A10.jpg	Não	2	Não se aplica
11	007A11.jpg	Não	0,5	Não se aplica
03	008A03.jpg	Não	2	Não se aplica
07	009A07.jpg	Não	1	Não se aplica
11	010A11.jpg	Não	2	Não se aplica
01	011A01.jpg	Não	0,5	Não se aplica
01	012A01.jpg	Não	3	Não se aplica
09	013A09.jpg	Não	1	Não se aplica
05	014A05.jpg	Não	2	Não se aplica
10	015A10.jpg	Não	0,5	Não se aplica
11	016A11.jpg	Não	1	Não se aplica
02	017A02.jpg	Não	2	Não se aplica
11	018A11.jpg	Não	2	Não se aplica
03	019A03.jpg	Não	2	Não se aplica
10	020A10.jpg	Não	1	Não se aplica
10	021A10.jpg	Não	0,5	Não se aplica
07	022A07.jpg	Não	1	Não se aplica
11	023A11.jpg	Não	2	Não se aplica
04	024A04.jpg	Não	3	Não se aplica
07	025A07.jpg	Não	1	Não se aplica
09	026A09.jpg	Não	2	Não se aplica
01	027A01.jpg	Não	3	Não se aplica
09	028A09.jpg	Não	2	Não se aplica
08	029A08.jpg	Não	2	Não se aplica
01	030A01.jpg	Não	1	Não se aplica
02	031A02.jpg	Não	2	Não se aplica
11	032A11.jpg	Não	1	Não se aplica
04	033A04.jpg	Não	2	Não se aplica
09	034A09.jpg	Não	2	Não se aplica
09	035A09.jpg	Não	1	Não se aplica
08	036A08.jpg	Não	1	Não se aplica

10	037A10.jpg	Sim	3	1
50	037A50P.jpg	Sim	3	1
10	038A10.jpg	Sim	2	2
16	038A16P.jpg	Sim	2	2
09	039A09.jpg	Sim	1	3
28	039A28P.jpg	Sim	1	3
04	040A04.jpg	Não	1	Não se aplica
04	041A04.jpg	Sim	1	1
34	041A34P.jpg	Sim	1	1
02	042A02.jpg	Sim	0,5	3
15	042A15P.jpg	Sim	0,5	3
11	043A11.jpg	Sim	1	3
41	043A41P.jpg	Sim	1	3
06	044A06.jpg	Sim	3	3
31	044A31P.jpg	Sim	3	3
11	045A11.jpg	Sim	1	1
36	045A36P.jpg	Sim	1	1
01	046A01.jpg	Sim	2	1
36	046A36P.jpg	Sim	2	1
03	047A03.jpg	Sim	2	3
20	047A20P.jpg	Sim	2	3
08	048A08.jpg	Não	1	Não se aplica
06	049A06.jpg	Sim	3	1
36	049A36P.jpg	Sim	3	1
01	050A01.jpg	Não	2	Não se aplica
09	051A09.jpg	Não	2	Não se aplica
13	052A13.jpg	Não	1	Não se aplica
13	053A13.jpg	Não	2	Não se aplica
12	054A12.jpg	Não	2	Não se aplica
12	055A12.jpg	Não	3	Não se aplica
12	056A12.jpg	Não	3	Não se aplica
14	057A14.jpg	Não	2	Não se aplica
15	058A15.jpg	Não	2	Não se aplica
15	059A15.jpg	Não	1	Não se aplica
14	060A14.jpg	Não	1	Não se aplica
16	061A16.jpg	Não	3	Não se aplica
14	062A14.jpg	Não	3	Não se aplica
21	063A21.jpg	Não	3	Não se aplica
16	064A16.jpg	Não	3	Não se aplica
13	065A13.jpg	Não	2	Não se aplica

13	066A13.jpg	Não	1	Não se aplica
19	067A19.jpg	Não	3	Não se aplica
12	068A12.jpg	Não	1	Não se aplica
19	069A19.jpg	Não	3	Não se aplica
17	070A17.jpg	Não	1	Não se aplica
16	071A16.jpg	Não	1	Não se aplica

I.2 CÓDIGO PYTHON DE EXTRAÇÃO DE IMAGENS

```

1
2 import os
3 import requests
4 from bs4 import BeautifulSoup
5 import re
6 from selenium import webdriver
7 from selenium.common.exceptions import NoSuchElementException
8 import time
9 import cssutils
10 import base64
11
12 # Os sites com as imagens de pessoas desaparecidas mostram as imagens de forma
13 #                               paginada,
14 # com um certo n mero X de imagens na primeira p gina, o mesmo n mero X de
15 #                               imagens na
16 # segunda p gina e assim por diante.
17 # Para baixar as imagens de forma automatizada, foram elaboradas duas alternativas
18 #                               :
19 # 1 - Para sites que mudam a url ao selecionar uma p gina no menu de pagina o :
20 #     * Realizar uma requisi o para cada url;
21 #     * Baixar as imagens atrav s da tag 'img' ou da propriedade 'background-
22 #                               image'
23 #     do css, dependendo do site.
24 # 2 - Para sites que n o mudam a url ao selecionar uma p gina, atualizando as
25 #                               imagens
26 #     de forma assncrona:
27 #     * Realizar uma requisi o para a url principal;
28 #     * Percorrer as p ginas clicando no menu de pagina o com o selenium;
29 #     * Baixar as imagens atrav s da tag 'img' ou da propriedade 'background-
30 #                               image'
31 #     do css, dependendo do site.
32 #
33 # Cada site precisa ser tratado separadamente, pois eles n o seguem um padr o.
34
35 # def extrairUrlBackgroundImg:

```

```

31 #     padrao = re.compile(r'url\("(.*?)"\)')
32 #
33 #     ocorrencias = padrao.findall(url)
34
35 # Extrai o dominio da url passada como parametro.
36 # Exemplo:
37 # url completa: https://www.desaparecidos.pr.gov.br/desaparecidos/desaparecidos.
38 #                 do?action=iniciarProcesso&m=true
39 # dominio retornado pela funcao: https://www.desaparecidos.pr.gov.br
40 # Serve para montar a url completa para cada imagem.
41 # Exemplo:
42 # atributo 'src' da tag 'img': /desaparecidos/exibirImagem.jpeg?codA=289190487&
43 #                               anoA=2022
44 # url completa da imagem 'dominio + src': https://www.desaparecidos.pr.gov.br/
45 #                                         desaparecidos/exibirImagem.jpeg?codA=
46 #                                         289190487&anoA=2022
47
48 #
49
50 def pageScroll(driver):
51     SCROLL_PAUSE_TIME = 2
52
53     # Get scroll height
54     last_height = driver.execute_script("return document.body.scrollHeight")
55
56     while True:
57         # Scroll down to bottom
58         driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
59
60         # Wait to load page
61         time.sleep(SCROLL_PAUSE_TIME)
62
63         # Calculate new scroll height and compare with last scroll height
64         new_height = driver.execute_script("return document.body.scrollHeight")
65         if new_height == last_height:
66             break
67         last_height = new_height
68
69
70 def extrairDominio(url):
71     padrao = re.compile(r'(https?:\/\/\..*?)\/')
72     # padrao = re.compile(r'(https?:\/\/www\..*?)\/')
73
74     ocorrencias = padrao.findall(url)
75
76     # print('Dominio extraido:', ocorrencias[0])
77
78     if not ocorrencias:
79         return None
80     else:
81         return ocorrencias[0]

```

```

79 # Verifica se a url est completa ou se possui somente o caminho sem o dominio.
80 # Exemplo de url completa: https://www.desaparecidos.pr.gov.br/desaparecidos/
                                exibirImagem.jpeg?codA=289190487&anoA=
                                2022
81 # Somente dominio: https://www.desaparecidos.pr.gov.br/
82 # Somente caminho: desaparecidos/exibirImagem.jpeg?codA=289190487&anoA=2022
83 def isUrlCompleta(url: str):
84     dominio = extrairDominio(url)
85
86     if dominio is None:
87         return False
88     else:
89         return True
90
91 def dowloadUsandoBackgroundImg(url: str, html: str, tag_class: str, nome_diretorio
                                : str, padrao_nome_img: str,
                                numero_img_inicial: int):
92     nome_diretorio = 'imagens/' + nome_diretorio
93
94     # Parse the HTML content with BeautifulSoup and find all image tags
95     soup = BeautifulSoup(html, 'html.parser')
96     img_tags = soup.find_all('div', class_=tag_class)
97
98     # Create a folder to store the images
99     if not os.path.exists(nome_diretorio):
100         os.makedirs(nome_diretorio)
101
102     # Download each image to the folder
103     print('URL:', url)
104     dominio = extrairDominio(url)
105     print('DOMINIO', dominio)
106     numero_img = numero_img_inicial
107     for img_tag in img_tags:
108         tag_property_style = img_tag['style']
109         style = cssutils.parseStyle(tag_property_style)
110         img_url = style['background-image']
111         img_url = img_url.replace('url(', '').replace(')', '')
112         if 'facebook' in img_url or 'logo' in img_url or 'png' in img_url:
113             continue
114         # img_name = img_url.split('/')[-1]
115         img_name = padrao_nome_img + str(numero_img) + '.jpg'
116         numero_img = numero_img + 1
117         img_path = os.path.join(nome_diretorio, img_name)
118         with open(img_path, 'wb') as f:
119             if isUrlCompleta(img_url):
120                 print(img_url)
121                 img_response = requests.get(img_url)
122             else:
123                 print(img_url)
124                 print(dominio+img_url)
125                 img_response = requests.get(dominio+img_url)
126             f.write(img_response.content)

```

```

127         print(f'Downloaded {img_name}')
128
129     qtd_imgs_baixadas = numero_img - numero_img_inicial
130
131     return qtd_imgs_baixadas
132
133 def dowloadUsandoTagImg(url: str, html: str, nome_diretorio: str, padrao_nome_img:
                                str, numero_img_inicial: int):
134     nome_diretorio = 'imagens/' + nome_diretorio
135
136     # Parse the HTML content with BeautifulSoup and find all image tags
137     soup = BeautifulSoup(html, 'html.parser')
138     img_tags = soup.find_all('img')
139
140     # Create a folder to store the images
141     if not os.path.exists(nome_diretorio):
142         os.makedirs(nome_diretorio)
143
144     # Download each image to the folder
145     print('URL:', url)
146     dominio = extrairDominio(url)
147     print('DOMINIO', dominio)
148     numero_img = numero_img_inicial
149     for img_tag in img_tags:
150         img_url = img_tag['src']
151         if 'facebook' in img_url or 'logo' in img_url or 'WhatsApp' in img_url or
                                'bg_disquedenuncia' in img_url\
152             or 'marco-site' in img_url or 'banner' in img_url or 'Banner' in
                                img_url or 'themes' in
                                img_url\
153             or 'LOGO' in img_url or 'Brasao' in img_url or 'Logo' in img_url
                                or 'icon' in img_url\
154             or 'footer' in img_url or 'sem-foto' in img_url or 'desaparecidos-
                                white.svg' in img_url\
155             or 'image/svg' in img_url or '.svg' in img_url or 'podcast' in
                                img_url or 'redesocial-
                                ' in img_url\
156             or 'popup' in img_url or 'component' in img_url:
157         print('URL com termo ignorado.')
158         continue
159         # img_name = img_url.split('/')[-1]
160         img_name = definirNomeImagem(padrao_nome_img, numero_img, img_url)
161         if 'extensao_nao_identificada' in img_name:
162             print('! Extensao n o identificada !')
163             continue
164         numero_img = numero_img + 1
165         img_path = os.path.join(nome_diretorio, img_name)
166         with open(img_path, 'wb') as f:
167             if 'base64' in img_url:
168                 print('Imagem base64')
169                 img_base64 = extrairStringImgBase64(img_url)
170                 img = base64.b64decode(img_base64)

```

```

171         elif isUrlCompleta(img_url):
172             print(img_url)
173             img_response = requests.get(img_url)
174             img = img_response.content
175         else:
176             print(img_url)
177             print(dominio+img_url)
178             try:
179                 img_response = requests.get(dominio+img_url)
180             except Exception as e:
181                 print(e)
182                 continue
183             img = img_response.content
184             f.write(img)
185             print(f'Downloaded {img_name}')
186
187     qtd_imgs_baixadas = numero_img - numero_img_inicial
188
189     return qtd_imgs_baixadas
190
191 def extrairStringImgBase64(img_url):
192     padrao = re.compile(r'data:image\/\.;base64,\s?(.)')
193
194     ocorrencias = padrao.findall(img_url)
195
196     if not ocorrencias:
197         return None
198     else:
199         return ocorrencias[0]
200
201 def definirNomeImagem(padrao_nome_img, numero_img, img_url):
202     if 'jpg' in img_url or 'jpeg' in img_url:
203         img_name = padrao_nome_img + str(numero_img) + '.jpg'
204     elif 'png' in img_url:
205         img_name = padrao_nome_img + str(numero_img) + '.png'
206     elif 'jfif' in img_url:
207         img_name = padrao_nome_img + str(numero_img) + '.jfif'
208     elif 'svg' in img_url:
209         img_name = padrao_nome_img + str(numero_img) + '.svg'
210     elif 'bmp' in img_url:
211         img_name = padrao_nome_img + str(numero_img) + '.bmp'
212     else:
213         img_name = padrao_nome_img + str(numero_img) + '.extensao_nao_identificada'
214
215         print('!Extens o n o identificada!')
216         print(img_url)
217
218     return img_name
219
220 def obterImagensDesaparecidosAM():
221     url = 'http://www.policiacivil.am.gov.br/desaparecidos/pag/'

```

```

222     nome_diretorio = 'AM-pc'
223     padrao_nome_img = 'desaparecidoAM-'
224     numero_paginas = 52
225
226     numero_img_inicial = 1
227     for numero_pagina in range(1, numero_paginas + 1):
228         url_completa = url + str(numero_pagina)
229         resposta = requests.get(url_completa)
230         html = resposta.content
231         qtd_imgs_baixadas = downloadUsandoTagImg(url_completa, html, nome_diretorio
232                                             , padrao_nome_img,
233                                             numero_img_inicial)
234
235         numero_img_inicial = numero_img_inicial + qtd_imgs_baixadas
236
237 def obterImagensDesaparecidosBA():
238     url = 'https://disquedenuncia.com/desaparecidos/page/'
239     nome_diretorio = 'BA-disquedenuncia'
240     padrao_nome_img = 'desaparecidoBA-'
241     numero_paginas = 38
242
243     numero_img_inicial = 1
244     for numero_pagina in range(1, numero_paginas + 1):
245         url_completa = url + str(numero_pagina)
246         resposta = requests.get(url_completa)
247         html = resposta.content
248         qtd_imgs_baixadas = downloadUsandoTagImg(url_completa, html, nome_diretorio
249                                             , padrao_nome_img,
250                                             numero_img_inicial)
251
252         numero_img_inicial = numero_img_inicial + qtd_imgs_baixadas
253
254 def obterImagensDesaparecidosDF():
255     url = 'https://www.pcdf.df.gov.br/servicos/desaparecidos'
256     nome_diretorio = 'pcdf'
257     padrao_nome_img = 'desaparecidoDF-'
258
259     downloadUsandoTagImg(url, nome_diretorio, padrao_nome_img)
260
261 def obterImagensDesaparecidosES():
262     url = 'https://disquedenuncia181.es.gov.br/desaparecidos#'
263     nome_diretorio = 'ES-disquedenuncia'
264     padrao_nome_img = 'desaparecidoES-'
265
266     driver = webdriver.Chrome('chromedriver.exe')
267
268     numero_img_inicial = 1
269     driver.get(url)
270     time.sleep(15)
271     while True:
272         html = driver.page_source
273         qtd_imgs_baixadas = downloadUsandoTagImg(url, html, nome_diretorio,
274                                             padrao_nome_img,
275                                             numero_img_inicial)

```

```

268     numero_img_inicial = numero_img_inicial + qtd_imgs_baixadas
269     try:
270         elemento = driver.find_element_by_link_text('Pr xima')
271         driver.execute_script("arguments[0].click();", elemento)
272         time.sleep(15)
273     except NoSuchElementException as e:
274         # print(e)
275         break;
276
277 def obterImagensDesaparecidosGO():
278     url = 'https://www.go.gov.br/servicos-digitais/dgpc/consulta-de-pessoas-
           desaparecidas-e-cadaveres-nao-
           identificados'
279     nome_diretorio = 'GO-ssp'
280     padrao_nome_img = 'desaparecidoGO-'
281
282     driver = webdriver.Chrome('chromedriver.exe')
283
284     numero_img_inicial = 1
285     driver.get(url)
286     time.sleep(10)
287     driver.find_element_by_xpath('//*[@id="content"]/div[2]/div/app-botao-acessar/
           div/button').click()
288
289     time.sleep(10)
290     while True:
291         html = driver.page_source
292         qtd_imgs_baixadas = downloadUsandoTagImg(url, html, nome_diretorio,
           padrao_nome_img,
           numero_img_inicial)
293         numero_img_inicial = numero_img_inicial + qtd_imgs_baixadas
294     try:
295         driver.find_element_by_xpath("//button[@aria-label='Pr ximo']").click
           ()
296         # driver.execute_script("arguments[0].click();", elemento)
297         time.sleep(5)
298     except NoSuchElementException as e:
299         print(e)
300         break;
301
302 def obterImagensDesaparecidosMT():
303     url = 'https://desaparecidos.pjc.mt.gov.br/inicio'
304     nome_diretorio = 'MT-pc'
305     padrao_nome_img = 'desaparecidoMT-'
306
307     driver = webdriver.Chrome('chromedriver.exe')
308
309     numero_img_inicial = 1
310     driver.get(url)
311     time.sleep(10)
312     while True:
313         html = driver.page_source

```

```

313     qtd_imgs_baixadas = dowloadUsandoTagImg(url, html, nome_diretorio,
                                                padrao_nome_img,
                                                numero_img_inicial)
314     numero_img_inicial = numero_img_inicial + qtd_imgs_baixadas
315     try:
316         driver.find_element_by_xpath('/html/body/app-root/div/div/app-inicio/
                                        div[2]/div[2]/pagination-
                                        controls/pagination-
                                        template/ul/li[10]/a').
                                        click()

317         time.sleep(5)
318     except NoSuchElementException as e:
319         print(e)
320         break;
321
322 def obterImagensDesaparecidosMG():
323     url = 'https://desaparecidos.policiacivil.mg.gov.br/desaparecido/album'
324     nome_diretorio = 'imagens/MG-pc'
325     padrao_nome_img = 'desaparecidoMG-'
326     numero_img = 1
327
328     resposta = requests.get(url)
329     html = resposta.content
330     soup = BeautifulSoup(html, 'html.parser')
331     padrao = re.compile(r'\\arquivo\\downloadArquivo\\/\\d*')
332     divs_imgs = soup.find_all('div', {'style': padrao})
333     urls_imgs = padrao.findall(str(divs_imgs))
334     dominio = extrairDominio(url)
335     for url_img in urls_imgs:
336         url_img_completa = dominio + url_img
337         img_name = padrao_nome_img + str(numero_img) + '.jpg'
338         numero_img = numero_img + 1
339         img_path = os.path.join(nome_diretorio, img_name)
340         with open(img_path, 'wb') as f:
341             print(url_img_completa)
342             img_response = requests.get(url_img_completa)
343             img = img_response.content
344             f.write(img)
345             print(f'Downloaded {img_name}')
346
347     # numero_img_inicial = 1
348     # resposta = requests.get(url)
349     # html = resposta.content
350     # dowloadUsandoBackgroundImg(url, html, nome_diretorio, padrao_nome_img,
351                                 numero_img_inicial)
352
353     # for numero_pagina in range(1, numero_paginas + 1):
354     #     url_completa = url + str(numero_pagina)
355     #     resposta = requests.get(url_completa)
356     #     html = resposta.content
357     #     qtd_imgs_baixadas = dowloadUsandoTagImg(url_completa, html,
358                                                 nome_diretorio, padrao_nome_img,

```

```

357         numero_img_inicial)
358
359     #     numero_img_inicial = numero_img_inicial + qtd_imgs_baixadas
360
361     def obterImagensDesaparecidosSC():
362         url = 'https://desaparecidos.pc.sc.gov.br/#/'
363         nome_diretorio = 'SC-pc'
364         padrao_nome_img = 'desaparecidoSC-'
365         numero_paginas = 52
366         tag_class = 'photo-wrapper'
367
368         driver = webdriver.Chrome('chromedriver.exe')
369
370         numero_img_inicial = 1
371         driver.get(url)
372         time.sleep(5)
373         for numero_pagina in range(1, numero_paginas + 1):
374             if numero_pagina != 1:
375                 elemento_filho = driver.find_element_by_xpath('//span[text()=' ' + str(
376                     numero_pagina) + '"]')
377                 elemento_pai = elemento_filho.find_element_by_xpath('..')
378                 elemento_pai.click()
379                 time.sleep(3)
380                 html = driver.page_source
381                 qtd_imgs_baixadas = downloadUsandoBackgroundImg(url, html, tag_class,
382                     nome_diretorio, padrao_nome_img
383                     , numero_img_inicial)
384                 numero_img_inicial = numero_img_inicial + qtd_imgs_baixadas
385
386     # def navegarPaginadorPeloTexto(textoLinkPaginador: str):
387     #     driver.find_element_by_link_text(textoLinkPaginador).click()
388
389     def obterImagensDesaparecidosPR():
390         url_menores_12 = 'https://www.desaparecidos.pr.gov.br/desaparecidos/
391             desaparecidos.do?action=
392             iniciarProcesso&m=true'
393
394         nome_diretorio_menores_12 = 'PR-pc/menores12'
395         padrao_nome_img_menores_12 = 'desaparecidoMenorPR-'
396         numero_paginas_menores_12 = 4
397
398         url_maiores_12 = 'https://www.desaparecidos.pr.gov.br/desaparecidos/
399             desaparecidos.do?action=
400             iniciarProcesso&m=false'
401
402         nome_diretorio_maiores_12 = 'PR-pc/maiores12'
403         padrao_nome_img_maiores_12 = 'desaparecidoMaiorPR-'
404         numero_paginas_maiores_12 = 357
405
406         driver = webdriver.Chrome('chromedriver.exe')
407
408         numero_img_inicial = 1
409         driver.get(url_menores_12)
410         time.sleep(5)
411         for pagina in range(1, numero_paginas_menores_12 + 1):

```

```

401     if pagina != 1:
402         driver.find_element_by_link_text(str(pagina) ).click()
403         time.sleep(5)
404         html = driver.page_source
405         qtd_imgs_baixadas = downloadUsandoTagImg(url_menores_12, html,
                                                    nome_diretorio_menores_12,
                                                    padrao_nome_img_menores_12,
                                                    numero_img_inicial)
406         numero_img_inicial = numero_img_inicial + qtd_imgs_baixadas
407
408     numero_img_inicial = 1
409     driver.get(url_maiores_12)
410     time.sleep(5)
411     for pagina in range(1, numero_paginas_maiores_12 + 1):
412         if pagina != 1:
413             driver.find_element_by_link_text(str(pagina) ).click()
414             time.sleep(5)
415             html = driver.page_source
416             qtd_imgs_baixadas = downloadUsandoTagImg(url_maiores_12, html,
                                                        nome_diretorio_maiores_12,
                                                        padrao_nome_img_maiores_12,
                                                        numero_img_inicial)
417             numero_img_inicial = numero_img_inicial + qtd_imgs_baixadas
418
419     # downloadUsandoTagImg(url_menores_12, nome_diretorio_menores_12,
420                             padrao_nome_img_menores_12)
421
422 def obterImagensDesaparecidosRJ():
423     url = 'https://desaparecidos.pcivil.rj.gov.br/pesquisar'
424     nome_diretorio = 'RJ-pc'
425     padrao_nome_img = 'desaparecidoRJ-'
426     numero_paginas = 16
427
428     driver = webdriver.Chrome('chromedriver.exe')
429
430     numero_img_inicial = 1
431     driver.get(url)
432     time.sleep(5)
433     for numero_pagina in range(1, numero_paginas + 1):
434         if numero_pagina != 1:
435             elemento = driver.find_element_by_xpath("//button[@aria-label='
436                                                         Pr xima p gina']")
437             driver.execute_script("arguments[0].click();", elemento)
438             time.sleep(5)
439             html = driver.page_source
440             qtd_imgs_baixadas = downloadUsandoTagImg(url, html, nome_diretorio,
                                                        padrao_nome_img,
                                                        numero_img_inicial)
441             numero_img_inicial = numero_img_inicial + qtd_imgs_baixadas
442
443 def obterImagensDesaparecidosRS():
444     url = 'https://www.pc.rs.gov.br/desaparecidos'

```

```

443     nome_diretorio = 'RS-pc'
444     padrao_nome_img = 'desaparecidoRS-'
445     numero_img_inicial = 1
446
447     driver = webdriver.Chrome('chromedriver.exe')
448     driver.get(url)
449     time.sleep(5)
450     pageScroll(driver)
451     html = driver.page_source
452     # resposta = requests.get(url)
453     # html = resposta.content
454     dowloadUsandoTagImg(url, html, nome_diretorio, padrao_nome_img,
455                             numero_img_inicial)
456
457 def obterImagensTodosDesaparecidos():
458     # AC n o tem site.
459     # AL utiliza o instagram.
460     # AP n o tem site.
461     # obterImagensDesaparecidosAM()
462     # obterImagensDesaparecidosBA()
463     # CE utiliza o instagram.
464     # obterImagensDesaparecidosDF()
465     # obterImagensDesaparecidosES()
466     # obterImagensDesaparecidosGO()
467     # MA
468     # obterImagensDesaparecidosMT()
469     # MS
470     # obterImagensDesaparecidosMG()
471     # PA
472     # PB
473     # PE
474     # PI n o tem site.
475     # obterImagensDesaparecidosPR()
476     # obterImagensDesaparecidosRJ()
477     # RN n o tem site.
478     obterImagensDesaparecidosRS()
479     # obterImagensDesaparecidosSC()
480
481 obterImagensTodosDesaparecidos()

```