

UNIVERSIDADE FEEVALE

SUELEM KLEINKAUF

PROTÓTIPO DE GERAÇÃO DE CARICATURAS BASEADA EM
FOTOS

Novo Hamburgo

2021

SUELEM KLEINKAUF

PROTÓTIPO DE GERAÇÃO DE CARICATURAS BASEADA EM
FOTOS

Trabalho de Conclusão de Curso apresentado
como requisito parcial à obtenção do grau
de Bacharel em Ciência da Computação pela
Universidade Feevale

Orientador: Dra. Marta Rosecler Bez

Novo Hamburgo

2021

SUELEM KLEINKAUF

PROTÓTIPO DE GERAÇÃO DE CARICATURAS BASEADA EM
FOTOS

Trabalho de Conclusão de Curso apresentado
como requisito parcial à obtenção do grau
de Bacharel em Ciência da Computação pela
Universidade Feevale

APROVADO EM: ____ / ____ / ____

DRA. MARTA ROSECLER BEZ
Orientadora – Feevale

GABRIEL DA SILVA SIMÕES
Examinador interno – Feevale

JOÃO BATISTA MOSSMANN
Examinador interno – Feevale

Novo Hamburgo
2021

AGRADECIMENTOS

Gostaria de agradecer a todos que, de alguma forma, contribuíram para a realização desse Trabalho de Conclusão de Curso. Em especial:

Minha família pelo apoio e compreensão. Pai e irmão, Sandro e Jeferson, por estarem sempre presentes me apoiando em cada passo.

Minha orientadora, Marta Rosecler Bez, por me guiar nesse percurso, mas para além disso, ser tia, ser apoio, ser inspiração. Por não me deixar desistir.

Aos meus amigos, que entenderam as ausências e me aguentaram em cada surto;

À Renata Dias, por ser compreensão e paciência até nos dias mais dificeis, sem ela, nada disso seria possível.

Ao professor Gabriel Simões e à Gustavo Cassel, por toda a ajuda e paciência em ensinar.

Aos professores que tive durante a vida, que me permitiram chegar onde cheguei com seus ensinamentos e doação!

Por fim, dedico este trabalho em memória de Marta de Paula, minha mãe, que acompanha do céu o fim dessa trajetória, mas que foi inspiração e apoio em todos os momentos. Gratidão por ser tua filha.

A todos os envolvidos, muito obrigado.

RESUMO

A Criatividade Computacional é uma área da computação que estuda a capacidade criativa de agentes computacionais, utilizando técnicas de inteligência artificial. Os avanços desse campo de pesquisa têm permitido a aplicação de seus métodos em diferentes áreas criativas, como, por exemplo, as artes visuais, na geração de pinturas e de imagens artísticas. Outra possibilidade para a aplicação de técnicas de Criatividade Computacional, dentro das artes, é a geração de caricaturas. As caricaturas são uma forma de retrato, em que são utilizadas técnicas de distorção ou de exagero, a fim de salientar as características principais do que está sendo representado, tornando o retrato mais facilmente identificável. Nesse cenário, esta pesquisa tem como objetivo desenvolver e validar um protótipo para a geração de caricaturas baseadas em fotos. A metodologia utilizada foi a *Design Search Research*, simplificada em 5 etapas: (1) identificação do problema, realizada por meio do estudo do campo da Criatividade Computacional, bem como da área na qual o presente estudo será aplicado, as caricaturas; (2) identificação dos artefatos, onde foi realizada uma revisão sistemática das técnicas e dos trabalhos que vêm sendo desenvolvidos nesse contexto; (3) desenvolvimento do protótipo, utilizando os *outputs* das etapas anteriores; (4) discussão de resultados, em que foram discutidas as imagens geradas; (5) comunicação dos resultados encontrados, realizada por intermédio da apresentação do trabalho e da publicação de artigos. Assim, foi realizado o treinamento de uma rede geradora de arquitetura CycleGan, utilizando o *dataset* Webcaricature, desse modo, as imagens satisfatórias foram geradas. Os resultados, salientam a utilidade desse tipo de rede neural para a geração de caricaturas, bem como demonstram possibilidades de estudos futuros dentro desse contexto.

Palavras-chave: Criatividade Computacional. Caricatura. *Design Science Research*.

ABSTRACT

Computational Creativity is an area of computing that studies the creative capacity of computational agents, using artificial intelligence techniques. Advances in this field of research have allowed the application of its methods in different creative areas, such as the visual arts, in the generation of paintings and artistic images. Another possibility for the application of Computational Creativity techniques within the arts is the generation of caricatures. Caricatures are a form of portrait, in which distortion or exaggeration techniques are used in order to highlight the main characteristics of what is being represented, becoming the portrait more easily identifiable. In this scenario, this research aims to develop and to validate a prototype for the generation of caricatures based on photos. The methodology used was Design Search Research, simplified into five steps: (1) identification of the problem, accomplished by means of the study of the field of Computational Creativity, as well as the area in which this study will be applied, the caricatures; (2) identification of artifacts, in which a systematic review of the techniques and the work that has been developed in this context was carried out; (3) prototype development, using the outputs from the previous steps; (4) discussion of results, in which the generated images were discussed; (5) communication of the results found, carried out by means of the presentation of the work and the publication of articles. Thus, the training of a generating network called CycleGan was conducted, using the Webcaricature dataset and, in this way, satisfactory images were generated. The results underline the usefulness of this type of neural network for the generation of caricatures, as well as they demonstrate possibilities for future studies within this context.

Keywords: Computational Creativity. Caricature. Design Science Research.

LISTA DE ILUSTRAÇÕES

Figura 1 – Etapas do trabalho	12
Figura 2 – Pintura gerada AARON	16
Figura 3 – Aplicação da técnica <i>Neural Style Transfer</i>	17
Figura 4 – Estudos do Pintor Annibale Carraci	19
Figura 5 – La Poire, Philipon	20
Figura 6 – Caricatura do Jornalista Carlos Lacerda	21
Figura 7 – Imagem x Caricatura Brad Pitt	22
Figura 8 – Caricaturas de Dilma Rousseff e Barack Obama	23
Figura 9 – Passo a passo da desconstrução da imagem	23
Figura 10 – Caricatura gerada pelo aplicativo MomentCam	24
Figura 11 – Caricatura gerada pelo aplicativo Cartoon Face	25
Figura 12 – Caricatura gerada pelo aplicativo Clip2Comic	25
Figura 13 – Caricatura gerada pelo aplicativo Face Caricature - Cartoon Photo	26
Figura 14 – Caricatura gerada pelo aplicativo Caricature Me	26
Figura 15 – Fluxograma de etapas de leitura com resultados	29
Figura 16 – Exemplos de imagens da base de dados	33
Figura 17 – Exemplo de arquitetura de modelo de rede adversária generativa	35
Figura 18 – Estrutura de DCGAN	35
Figura 19 – Exemplo de arquitetura de modelo de rede adversária generativa	37
Figura 20 – Arquitetura do modelo CycleGan	38
Figura 21 – Resultado do treinamento da arquitetura DCGAN	39
Figura 22 – Funções para adequação do formato dos dados	40
Figura 23 – Exemplo de antes e depois do processo de retirada dos detalhes de fundo	42
Figura 24 – Resultados aleatórios gerados com treinamento de 100 épocas	43
Figura 25 – Utilização do modelo de 100 épocas	43
Figura 26 – Treinamento com 100 épocas e com hiperparâmetro batch_size 4	44
Figura 27 – Comparação de <i>Loss</i> entre treinamentos de batch_size diferentes por época	45
Figura 28 – Utilização do modelo de 100 épocas com batch_size 4	45
Figura 29 – Resultados gerados durante treinamento com 150 épocas	46
Figura 30 – Aplicação do modelo de 150 épocas	47
Figura 31 – Imagens geradas pelo treinamento usando imagens com fundo neutro	48
Figura 32 – Aplicação do modelo treinado com imagens pré processadas	48

LISTA DE TABELAS

Tabela 1 – Artigos por ano de publicação, título e autor	30
--	----

LISTA DE ABREVIATURAS E SIGLAS

ACC	<i>Association for Computational Creativity</i>
CLM	<i>Constrained Local Models</i>
CQF	<i>Convex Quadratic Fitting</i>
DCGAN	<i>Deep Convolutional Generative Adversarial Network</i>
ELM	<i>Extreme Learning Machine</i>
GANs	<i>Conditional Generative Adversarial Networks</i>
IA	Inteligência Artificial
LLE	<i>Locally Linear Embedding</i>
StArt	<i>software State of the Art through Systematic Review</i>

SUMÁRIO

1	Introdução	10
2	Metodologia	12
3	Criatividade Computacional	14
4	Caricatura	18
5	Revisão Sistemática	28
5.1	Quais as técnicas utilizadas no estudo?	30
5.2	Que base de dados foi utilizada para treinamento e testes?	31
5.3	Como o trabalho foi validado?	31
5.4	Que características faciais foram utilizadas?	32
6	Desenvolvimento	33
6.1	Base de dados utilizada	33
6.2	Redes Adversárias Generativas (GANs)	34
6.3	DCGAN	35
6.4	CycleGAN	36
6.5	Treinamento	38
6.6	Resultados	42
6.6.1	Treinamento de 100 épocas	42
6.6.2	Treinamento de 100 épocas com <i>batch_size</i> 4	44
6.6.3	Treinamento de 150 épocas	46
6.6.4	Treinamento com imagens com fundo neutro	47
7	Conclusão	50
	Referências	52
	APÊNDICES	57
	APÊNDICE A Formulário de solicitação do <i>dataset</i>	58
	APÊNDICE B E-mail de autorização para uso do <i>dataset</i>	59
	APÊNDICE C Tabela de <i>Loss</i> treinamento com 100 épocas	60
	APÊNDICE D Tabela de <i>Loss</i> treinamento com 100 épocas e <i>batch_size</i> 4	62
	APÊNDICE E Tabela de <i>Loss</i> treinamento com 150 épocas	65
	APÊNDICE F Tabela de <i>Loss</i> treinamento com imagens neutralizadas	67

1 INTRODUÇÃO

O termo *Criatividade Computacional*, definido por Pereira (2007), descreve uma área da computação que desenvolve sistemas que apresentam um comportamento criativo baseado em fundamentos computacionais, os quais são capazes de criar artefatos. Trata-se de um recente campo de pesquisa, o qual utiliza técnicas da Inteligência Artificial, buscando a modelagem e a simulação da criatividade humana, por meio de mecanismos capazes de apresentar soluções inovadoras para diferentes problemas (BRITO; VANZIN; ULRICH, 2009).

De acordo com Veronese (2017), diversas pesquisas têm buscado o desenvolvimento de agentes, robôs ou *softwares* que não sejam simples ferramentas, mas capazes de produzir soluções de problemas de forma original e criativa. O desenvolvimento de um mecanismo que demonstre algum tipo de processo criativo pode beneficiar diversas áreas, como as artes, a pintura, a música, os jogos, dentre outros (VELLAR, 2019).

Dentro das artes, por exemplo, uma das muitas possibilidades para as técnicas da Criatividade Computacional é a geração de caricaturas de forma automática ou semi-automática, ajudando o artista na criação. Segundo Sadimon *et al.* (2010), caricatura é uma representação pictórica e literária de uma pessoa ou de uma coisa, na qual se busca exagerar características que tenham uma semelhança visual facilmente identificável. Para Mauro e Kubovy (1992), a caricatura, assim como outros tipos de retratos, busca apresentar pontos da essência da pessoa representada.

Segundo os mesmos autores, as caricaturas, quando bem-sucedidas, podem despertar o reconhecimento melhor do que retratos fidedignos. Isso acontece porque, nas caricaturas, as características distintivas são exageradas, o que enfatiza as características que são percebidas pelo observador com mais facilidade, mesmo em um curto período de tempo, ajudando, assim, no reconhecimento. Devido a isso, tais representações são comumente utilizadas pela polícia em investigações ou para identificação de sujeitos no processo forense (SADIMON *et al.*, 2010). Por isso, a geração de uma caricatura precisa levar em consideração os pontos-chave da face na qual será retratada.

Nesse cenário, este trabalho tem como objetivo desenvolver um protótipo de geração automática ou semiautomática de caricaturas baseadas em fotos, utilizando as técnicas da Criatividade Computacional. Para isso, a presente pesquisa foi organizada em capítulos, os quais são descritos após esta Introdução.

O Capítulo 2, inicialmente, explica a Design Science Research, metodologia escolhida para este trabalho, simplificada em cinco etapas. O Capítulo 3, por sua vez, apresenta um estudo a respeito de Criatividade Computacional, bem como suas definições e algumas

das aplicações práticas das suas técnicas. Em seguida, o Capítulo 4, descreve os conceitos de caricatura, buscando entender o contexto de aplicação do presente estudo. Essas são as duas grandes áreas que transpassam esta pesquisa. No Capítulo 5, encontra-se a descrição de passos e as respostas geradas por uma revisão sistemática, a qual foi realizada buscando conhecer os trabalhos que vêm sendo desenvolvidos na área em questão, bem como as técnicas utilizadas e as formas de validação e de materiais.

Os resultados encontrados serviram de insumos para o desenvolvimento do protótipo em questão. Por conseguinte, no Capítulo 6, são explicados os artefatos utilizados, assim como o processo de treinamento da rede neural, para a geração de caricaturas. Além disso, esse capítulo apresenta os resultados alcançados. Por fim, o Capítulo 7 encerra o presente trabalho com a conclusão deste estudo.

2 METODOLOGIA

A metodologia deste trabalho é baseada nos conceitos apresentados pela *Design Science Research* (DSR). A DSR disponibiliza fundamentos para legitimar o desenvolvimento de artefatos como um meio de produção de conhecimentos científicos. Essa metodologia operacionaliza a base epistemológica *Design Science*, também conhecida como ciência de projeto (SIMON, 1996). Segundo Lacerda *et al.* (2013), a *Design Science Research* é definida como um processo rigoroso para desenvolver artefatos a fim de resolver problemas, avaliar o que foi desenvolvido e comunicar os resultados obtidos.

Os artefatos são definidos como algo artificial, que individualmente não soluciona o problema. Todavia, auxilia no processo de solução do problema (DRESCH; LACERDA; JÚNIOR, 2015). Para Jr, Chen e Purdin (1990) esses artefatos podem ser modelos, construtos, métodos, instanciações e sistemas de informações.

Neste trabalho, foram definidas cinco etapas indicadas pela metodologia, buscando adequar ao tempo e aos objetivos da presente pesquisa. A Figura 1 apresenta as cinco etapas realizadas, sendo elas: (i) a identificação do problema; (ii) a identificação dos artefatos; (iii) o desenvolvimento do protótipo; (iv) a discussão dos resultados; e, por fim, (v) a comunicação dos resultados encontrados.

Figura 1 – Etapas do trabalho



Fonte: Do autor (2021)

Com base na figura acima, cada uma das etapas será aprofundada a seguir, buscando apresentar um breve resumo do que foi realizado em cada uma delas.

Identificação do problema: nesta etapa, objetiva-se reconhecer e determinar o problema que o estudo buscou solucionar. No presente trabalho, foram realizados estudos das áreas de Criatividade Computacional e de caricaturas. Por meio destes estudos,

identificou-se a possibilidade da geração de caricaturas baseadas em fotos de forma automática, por meio de técnicas de inteligência artificial. Os estudos são encontrados nos Capítulo 3 e 4, respectivamente.

Identificação dos artefatos: uma vez que o problema foi identificado, torna-se possível encontrar artefatos já desenvolvidos por outros pesquisadores, os quais auxiliaram na resolução do problema em questão, além de propor novos artefatos, se necessário. Nesse sentido, foi realizada uma revisão sistemática, a qual permite ao pesquisador situar-se no contexto e conhecer os trabalhos, além da identificação dos artefatos. No protocolo inicial, foram definidos critérios e questões norteadoras que possibilitaram a definição do uso de redes geradoras, bem como, apresentaram o *dataset* utilizado no decorrer deste estudo. Esse processo é apresentado no Capítulo 5.

Desenvolvimento do protótipo: após as etapas anteriores, inicia-se o desenvolvimento da solução do problema levantado, utilizando os artefatos encontrados. No Capítulo 6, descreve-se o processo dessa construção. Quanto à tecnologia, foram utilizadas redes adversárias geradoras, conhecidas como GANs, além de outras ferramentas que auxiliaram em todo desenvolvimento, tais como TensorFlow, Keras e GoogleColab.

Discussão dos resultados: após o desenvolvimento da solução, foram realizadas a apresentação e a discussão resultados. Isso acontece para entender se o que foi gerado atende à ideia prevista na etapa de identificação do problema.

Comunicação dos resultados: esta é a etapa final da DSR, que busca disseminar o conhecimento construído com pessoas e organizações que tenham interesse no assunto em questão. Para isso, foi realizada uma apresentação deste trabalho na Feira de Iniciação Científica da Universidade Feevale. Além disso, os resultados são apresentados no presente trabalho, bem como, pretende-se publicar artigos científicos em congressos ou revistas.

Dito isso, no capítulo a seguir, são apresentados os estudos sobre Criatividade Computacional que serviram para a identificação do problema, sendo essa a primeira etapa desta pesquisa.

3 CRIATIVIDADE COMPUTACIONAL

Ao apresentar o termo *Criatividade Computacional*, é importante compreender os conceitos de criatividade nos quais este trabalho se baseia, a fim de nortear esta pesquisa de forma clara, já que o termo é um fenômeno que comprehende diferentes aspectos e funciona em diferentes escalas (PIRES, 2017).

Assim como na Inteligência Artificial, não há um consenso sobre o significado de criatividade. Todavia, alguns autores convergem na ideia de que, no processo criativo, ressignifique-se ou crie-se algo novo, implicando na reconstrução do passado e na reinterpretação do presente (BARRETTTO, 2016). Para Amabile (1996), um artefato, seja ele um produto, seja ele uma resposta a um problema, pode ser considerado criativo na medida em que é novo e apropriado, além de conter valor para o problema ou tarefa em questão.

Para o desenvolvimento deste estudo, toma-se como definição de criatividade o conceito de Amabile (1996), a qual destaca alguns aspectos importantes, como a originalidade do que está sendo apresentado e a adequação da resposta, como dito anteriormente. Segundo ela, para que se possa avaliar a criatividade de um processo, também é necessário que a tarefa proposta possibilite várias soluções como resposta.

As definições de Boden *et al.* (2004) reafirmam as ideias apresentadas por Amabile (1996), quando se referem à criatividade como a capacidade de apresentar ideias ou artefatos que além de novos são valiosos. As ideias incluem poemas, músicas, conceitos, receitas culinárias, entre outros, e os artefatos incluem pinturas, cerâmicas, origamis, dentre outras possibilidades.

Amabile (1998) destaca que o pensamento criativo se compõe das maneiras como são abordados os problemas e as soluções, além da capacidade do indivíduo de reunir as ideias existentes em novas combinações. Sendo assim, tem-se o seguinte questionamento: um artefato novo, gerado por um agente computacional e baseado em um conjunto de dados existentes, poderia ser considerado um artefato criativo? Boden *et al.* (2004) afirmam que os computadores podem gerar ideias e artefatos aparentemente criativos e, além disso, ajudar a compreender o processo da criatividade nas pessoas.

O termo Criatividade Computacional define um amplo campo de pesquisa que envolve técnicas de computação, principalmente a Inteligência Artificial (IA), em processos criativos de agentes computacionais. A área em questão é interdisciplinar, sendo aplicada em áreas como o Design, a Arquitetura, a Música, entre outros (BERGAMO; SILVA, 2020). Segundo Colton, Wiggins *et al.* (2012), a Criatividade Computacional pode ser definida como um subcampo da Inteligência Artificial, no qual se produz e se trabalha com sistemas computacionais que constroem artefatos e ideias.

Para definir a Criatividade Computacional, a *Association for Computational Creativity* (ACC) (2017), organização que promove anualmente o Congresso Internacional sobre Criatividade Computacional, descreve-o como um campo de pesquisa que está prosperando na formalização do que significa ser criativo em software. Ela apresenta sistemas computacionais, que em conjunto com outras áreas e assumindo responsabilidades específicas, exibem comportamentos que os observadores imparciais consideram criativos.

Segundo Vellar (2019), quando se fala em Criatividade Computacional, é a partir do desenvolvimento de algoritmos, utilizando conceitos de Inteligência Artificial, que começa-se a perceber indícios de criatividade por parte dos computadores. Nesse sentido, o *hardware* por si só é apenas uma máquina estática que não conseguiria gerar nada além do que fora programado para fazer.

Bergamo e Silva (2020) salientam que o processo criativo em si não é o objetivo, uma vez que a codificação e o desenvolvimento desse tipo de agente trabalha em conjunto com as outras áreas apresentadas, buscando a geração de um artefato original. Essa geração acontece por meio de um processo que além de propor, gera, avalia, realiza uma mudança na lógica criativa e, por fim, valida os resultados estabelecidos. Segundo Amabile (1996), um artefato pode ser considerado criativo, se for: novo, apropriado, útil, correto ou de valor para a tarefa (ou problema) em questão.

No evento da Campus Party, em 2019, Apolinário Passos realizou uma palestra a respeito da Criatividade Computacional, definindo a criatividade como uma filosofia de engenharias de sistemas computacionais que exibem comportamentos que podem ter diferenciais criativos e que são criados pelos humanos para que as máquinas possam operar, indicando, assim, que o processo é a criação humana ajudando a máquina a criar (IH!CRIEI, 2021). Nesse mesmo contexto, Colton, Wiggins *et al.* (2012) entendem que os sistemas computacionais não são humanos, e que a criatividade por eles apresentada será criatividade, mas não como nós a conhecemos em seres humanos.

Todavia, a ideia de apresentar um sistema criativo já existe a bastante tempo. Nas décadas de 1960 e 1970, quando os primeiros computadores começaram a surgir, as pessoas já pensavam em como construir sistemas que pudessem criar arte, música e literatura. E já na década de 1970, ocorreu o primeiro simpósio de Criatividade Computacional, que ainda era muito exploratório, mas que envolvia testes com máquinas e robôs (IH!CRIEI, 2021).

Nas décadas seguintes, a área teve avanços consideráveis com a criação do *software* AARON, em meados de 1973, o qual cria pinturas aleatórias e foi desenvolvido pelo artista e programador Harold Cohen (IH!CRIEI, 2021). O robô funciona a partir da inclusão de diferentes conhecimentos sobre pintura, como, por exemplo, sobre a mistura de tintas. Com base nesses conhecimentos, AARON combina as informações adquiridas e pode produzir diferentes desenhos usando um estilo próprio. Nesse sentido, AARON não

recorre a uma base de dados com pinturas prontas (VELLAR, 2019).

Segundo Apolinário (2018), nesse processo, o programador não descreve como será cada pintura, mas cria mecanismos para que o sistema possa tomar decisões, fazendo com que, mesmo que os *inputs* sejam os mesmos, as pinturas geradas sejam diferentes. A Figura 2 apresenta uma das pinturas realizadas pelo robô AARON.

Figura 2 – Pintura gerada AARON



Fonte: Cohen (2016) apud Apolinário (2018)

Desde então, o campo da Criatividade Computacional está em grande expansão, com a criação de novos sistemas computacionais que geralmente, não exclusivamente, são aplicados em áreas historicamente associadas à criatividade, como matemática e ciência, poesia, composição musical, *design* e até culinárias (VELLAR, 2019).

Um caso real de aplicação das técnicas de Criatividade Computacional na publicidade é apresentado por Kirkpatrick (2017). As empresas IBM e Toyota treinaram a famosa IA Watson para escrever as chamadas publicitárias do lançamento de seu novo carro, alimentando-o com chamadas que já haviam sido bem-sucedidas anteriormente. Após a geração bem sucedida, um redator finalizou o trabalho. Nitidamente, o processo de criação foi facilitado pela IA em questão (IBM, s/d).

O trabalho de Vellar (2019) também apresenta uma interessante aplicação da técnica de Criatividade Computacional com três agentes: (i) um humano que fornece as imagens de origem; (ii) outro humano com estilo artístico de pintura conhecido; e (iii) um agente de Inteligência Artificial (IA). A técnica utilizada foi a *Neural Style Transfer*, classificada como técnica co-criativa. Ela foi aplicada a uma imagem para a geração de pinturas em diferentes estilos artísticos, como o Cubismo, Surrealismo, Expressionismo e Futurismo. Na Figura 3, apresenta-se o resultado da aplicação da técnica descrita na imagem de origem.

Figura 3 – Aplicação da técnica *Neural Style Transfer*

Experimento	Imagen de origem	Estilo aplicado	Artefato Gerado
Cubismo 1			
Surrealismo 2			
Expressionismo 2			
Futurismo 1			

Fonte: Vellar (2019)

O resultado mostra-se bem sucedido quanto à transferência de estilos artísticos conhecidos para a imagem escolhida. Segundo IH!CRIEI (2021), existem diversas técnicas ligadas à Criatividade Computacional que utilizam-se dessa mesma ideia, em que, com base em um *input* de estilo, o agente as aprende e as aplica em outras imagens.

A IA, com suas diferentes técnicas, mostra-se ideal para fornecer assistência para muitos setores, como a saúde, as finanças e o varejo. Existem ainda limitações nos processos criativos, mas não há dúvida que eles serão ampliados cada vez mais. Apesar de nunca substituírem a criatividade humana, as técnicas de Criatividade Artificial podem oferecer grandes benefícios, servindo como assistente inteligente, eficiente e inspirador.

No próximo capítulo, são apresentados conceitos e técnicas de caricatura, a fim de entender o contexto em que as técnicas de Criatividade Computacional, investigadas neste trabalho, são aplicadas.

4 CARICATURA

Como descrito no capítulo anterior, as técnicas de Criatividade Computacional podem ser empregadas em diferentes áreas com diversas finalidades. Grande parte das aplicações encontradas na literatura são em áreas consideradas criativas e que permitem diferentes soluções possíveis. Nesse sentido, este trabalho tem como objetivo gerar caricaturas de forma automática ou semiautomática, utilizando técnicas de Inteligência Artificial.

As caricaturas, de forma geral, são definidas como uma forma divertida de desenhar pessoas, enfatizando suas características proeminentes (SIDDAVATAM; SIDDAVATAM, 2011). Li *et al.* (2020) também apresentam um conceito parecido do significado de caricatura, afirmando que é uma criação artística produzida com a exageração de algumas características salientes na imagem da face, preservando a identidade do sujeito ou coisa. Os autores afirmam, ainda, que as caricaturas são geralmente utilizadas em mídias sociais para diferentes propósitos, sendo um bom exemplo a utilização desse recurso para expressar sentimentos e emoções.

Para Hughes (1999), a caricatura é uma arte que busca capturar a essência da personalidade de uma pessoa por meio da semelhança exagerada, no intuito de criar um retrato mais fiel à vida do que a própria vida. Rhodes *et al.* (1998) reforçam essa ideia em experimentos que buscam comprovar que, por meio de caricaturas, é mais fácil reconhecer alguém do que por uma imagem ou fotografia, já que elas chamam a atenção para os detalhes que mais caracterizam o rosto desenhado.

Nas artes visuais, caricatura se refere a um estilo de retrato, ou seja, a criação de uma imagem de pessoa ou coisa com a intenção de capturar personalidade, humor e semelhança, em que a distorção é utilizada para tornar o sujeito retratado, mais facilmente identificado (SMITH; MATEAS, 2011). Uma caricatura de um ditador conhecido, por exemplo, geralmente irá muito além do exagero em sua fisionomia, mas também buscará expor muito de seu perfil psicológico.

Segundo Carazo (2003), em seu livro denominado *Caricaturas*, a palavra caricatura tem origem na palavra italiana *caricare*, que significa exagerar ou aumentar as proporções de algo. Nesse sentido, ela tem o propósito de exagerar as características de uma pessoa, acentuando aspectos segundo a intenção do artista. Conforme Strømsted (2017), o pintor italiano Annibale Carracci estabeleceu o termo *Caricatura* durante o século XVI, quando era particularmente conhecido por suas caricaturas humorísticas e exageradas que contrastavam com o idealismo contemporâneo. Alguns estudos realizados por ele são apresentados na Figura 4, na qual pode-se perceber uma preocupação em aproximar-se

da realidade e exagerar alguns detalhes das faces apresentadas, como narizes e queixos.

Figura 4 – Estudos do Pintor Annibale Carraci



Fonte: WikiArt (2011)

Mendes (2007) afirma que a caricatura funciona como um superretrato, sendo uma representação mais autêntica de uma pessoa do que sua fotografia sem qualquer manipulação. Isso se baseia na hipótese da peculiaridade, que é determinada pela comparação entre a pessoa em questão e o protótipo. Dessa maneira, como a caricatura enfatiza as peculiaridades encontradas, ela encontra-se mais próxima ao que está armazenado na memória. Gooch, Reinhard e Gooch (2004) afirmam que os humanos reconhecem rostos com base nas características faciais que se desviam de um rosto tido como padrão.

Apesar disso, os autores discordam sobre o nível em que os elementos de uma caricatura devem ser exagerados para que seu efeito seja maximizado (RHODES *et al.*, 1998) (LEE; BYATT; RHODES, 2000). O exagero de características faciais comuns de um sujeito pode levar à distorção da identidade da pessoa retratada; por isso, é necessário enfatizar de maneira correta seus traços físicos ou sua personalidade (LE; WHY; ASHRAF, 2011). Por outro lado, Smith e Mateas (2011) afirmam que o que torna uma caricatura notável é muito mais que os aspectos escolhidos para serem enfatizados, mas outras escolhas que o artista emprega, como, por exemplo, os elementos estilísticos, como espessura de linha ou sombreamento.

Para Le, Why e Ashraf (2011), caricaturas de rosto são normalmente utilizadas pela mídia impressa e também pela indústria de animação. Elas estão bastante presentes nas últimas décadas, seja como forma de entretenimento, apresentando estrelas de cinema,

seja trazendo expressão da perspectiva social e política, criticando a intolerância, injustiça e corrupção política usando humor e sátira. A representação caricatural de figuras políticas são normalmente encontradas em desenhos editoriais, já que resume as ideias de um artigo inteiro a imagens simples (STRØMSTED, 2017).

Strømsted (2017) afirma que a caricatura tem sido considerada uma importante fonte de material para jornalistas e historiadores devido ao seu uso, não somente como diversão inocente, mas pela metáfora e mensagem por ela apresentada. Smith e Mateas (2011) reafirmam essa ideia ao dizer que esse tipo de retrato pode expressar algumas ideias e reivindicações do artista mais rapidamente do que o equivalente verbal. As caricaturas políticas provaram ser um poderoso veículo para influenciar a opinião pública, elogiando ou criticando políticas e ações de governantes (SADIMON *et al.*, 2010).

Para Smith e Mateas (2011), quando constrói uma caricatura, cada artista faz uma declaração sobre o sujeito retratado e diferentes artistas podem decidir exagerar aspectos diferentes ou apresentar os mesmos aspectos de formas diferentes. Segundo Júnior (2016), o desenho caricatural é um produto cultural orientado por circunstâncias históricas, todavia, ele não deve ser considerado como algo inocente, pois, além do riso, a caricatura é capaz de, dentre outros fatores, denunciar abusos, firmar preconceitos e estereótipos, até desumanizar o outro e reforçar posições políticas.

Um dos exemplos históricos importantes da caricatura e que exemplifica bem a questão social é abordada por Filho (2006) e apresentada na Figura 5, denominada de *La Poire*, de Charles Philipon, datada do ano de 1827. Ela retrata o Rei Luiz Filipe como uma pêra, termo que se referia a pessoas sem inteligência.

Figura 5 – La Poire, Philipon



Fonte: Filho (2006)

O artista foi levado aos tribunais pela publicação, mas acabou absolvido ao demonstrar os passos realizados para chegar a esse resultado, comprovando que o rei realmente se parecia com o símbolo em questão. A figura da pêra passou a ser apresentada em

inúmeras caricaturas após a situação, já que se demonstrou um tipo de zombaria muito eficiente (FILHO, 2006). Esse cenário exemplifica que a caricatura não é definida por um desenho de humor, apenas passa por ele, e, por essa razão, significa mais do que a distração e graça que provoca. Ela apresenta conteúdos político e cultural, geralmente pela via da ridicularização e da ironia, defendendo posições políticas, enfatizando diferenças e atuando na realidade dos povos (JÚNIOR, 2016).

Outro exemplo histórico de caricatura, que é apresentado na Figura 6, foi desenhada por Lan e publicada no jornal *A Última Hora*, que retratava o jornalista Carlos Lacerda como um corvo, símbolo de mau agouro e de morte. O jornalista passou a ser relacionado ao animal pela imprensa contrária a ele e também pelos seus próprios colegas (ARRIGONI, 2011).

Figura 6 – Caricatura do Jornalista Carlos Lacerda



Fonte: Arrigoni (2011)

Na época, o jornalista foi acusado de estar presente sempre que ocorria uma morte interessante, buscando informações, fato esse que inspirou a caricatura em questão (CORRÊA, 2015). É possível observar, nessa caricatura outra técnica utilizada pelos desenhistas, a zoomorfia. A caricatura busca representar por meio do exagero em primeiro lugar, mas conforme Arrigoni (2011), mais do que apenas "carregar" os traços da pessoa retratada: ela busca caracterizar.

Machado (2013) apresenta dicas para a construção de caricaturas. Para ele, é

necessário, primeiramente, observar a pessoa que será retratada, tentando encontrar as principais características. Após isso, definir se o desenho será apenas de rosto ou de corpo. Ao iniciar o desenho, ele sugere começar pelos contornos, já exagerando o que a pessoa tem de mais chamativo (detalhes grandes e pequenos). O próximo passo é inserir os detalhes internos, buscando deixar apenas poucos traços.

Na Figura 7 o autor apresenta tanto uma análise de uma foto do ator Brad Pitt quanto a caricatura baseada na foto. Por intermédio, pode-se compreender quais os tipos de características são necessárias observar na construção da arte. Machado (2013) destaca que não se trata apenas do exagero, mas também pode-se reduzir componentes faciais que já são pequenos.

Figura 7 – Imagem x Caricatura Brad Pitt



Fonte: Machado (2013)

Caricaturas políticas ainda são bem comuns nos dias atuais, nas quais diversos representantes governamentais são retratados em jornais e revistas. Como visto anteriormente, a caricatura e a política sempre tiveram suas conexões, sendo ela usada para contestar decisões, apresentar opiniões e levar às classes menos favorecidas os acontecimentos, de forma mais entendível. Duas figuras muito representadas por meio de caricaturas foram Dilma Rousseff e Barack Obama, as quais, apesar da grande variedade disponível na internet, são normalmente similares às características exageradas. Em Dilma, os dentes e cabelo são sempre evidenciados; já em Obama, são as orelhas, queixo e o nariz que ganham destaque. Na Figura 8, são apresentados dois exemplos de caricaturas das figuras políticas citadas, feitas por Amarildocharge (2011), chargista e editor de ilustração do jornal A Gazeta de Vitória (ES).

Outra técnica de desenho de caricatura, indicada a quem está iniciando o processo de aprendizagem da arte, é a desconstrução da imagem. Nela parte-se de uma imagem

Figura 8 – Caricaturas de Dilma Rousseff e Barack Obama



Fonte: Amarildocharge (2011)

mais robusta, e com maior detalhamento. Seguindo etapas, vai-se simplificando e dando enfoque aos detalhes proeminentes na pessoa retratada. A Figura 9 apresenta as etapas da técnica de desconstrução.

Figura 9 – Passo a passo da desconstrução da imagem



Fonte: Mariani (2014)

Nesse sentido, segundo Mariani (2014), são consideradas 4 etapas do processo de desconstrução que darão origem à caricatura. São elas:

1. escolher uma foto e observá-la para que fique bem fixa em sua mente;
2. fazer um desenho de observação, o mais próximo possível da realidade; Considerando o tipo de cabelo, formato do rosto e tamanho dos elementos;
3. realizar um desenho mais descontraído, sem preocupação com os detalhes, mas mantendo as características mais evidentes; e
4. desenhar mais rápido, preocupando-se em exagerar as principais características da pessoa retratada.

Além das técnicas apresentadas, é possível, atualmente, utilizar aplicativos e sites desenvolvidos para esse fim. No site *TechTudo*, Fernandes (2019) apresenta uma lista de

10 aplicativos para transformar fotos em caricaturas. Para entender o funcionamento e os resultados, realizou-se o *download* de alguns dos aplicativos citados, por meio de um celular IOS. Alguns deles não foram encontrados com o nome que a matéria apresentava. Além disso, realizou-se uma busca na loja de aplicativos pelos termos *caricature* e *caricatura*, para que também fossem testados aplicativos que não estavam na lista acima.

Ao todo, foram seis aplicativos baixados e utilizados. Todos eles estão disponíveis na *App Store* de telefones IOS gratuitamente. Abaixo, apresenta-se a descrição do processo e as imagens geradas:

MomentCam: o aplicativo solicita uma foto de rosto e utiliza os componentes como olhos, boca e nariz para deixar um avatar com a aparência do usuário. Depois disso, pode-se escolher o sexo, a idade, além de algumas características como tipo e cor do cabelo. É possível mudar os demais elementos também, tornando o resultado mais próximo a um avatar desenhado e menos parecido com a foto inicial. Como apresentado na Figura 10, não há exagero de quaisquer características, mas a imagem lembra bem a pessoa retratada.

Figura 10 – Caricatura gerada pelo aplicativo MomentCam



Fonte: Do Autor (2021)

Mojipop: mesmo aplicativo que o *MomentCam*, disponibilizado com nome diferente. A caricatura gerada é a mesma da Figura 10.

Cartoon Face: após realizado o *upload* de uma imagem, o aplicativo gera uma caricatura com base nela. O resultado pode ser visualizado na Figura 11, sendo o que melhor representou a ideia de caricatura, trazendo uma semelhança à foto original. O próprio aplicativo coloca a foto original e a caricatura gerada lado a lado.

Clip2Comic: o aplicativo permite que seja realizado o *upload* de uma imagem

Figura 11 – Caricatura gerada pelo aplicativo Cartoon Face



Fonte: Do Autor (2021)

ou que seja tirada uma foto, para servir como base. Todavia, ele apenas disponibiliza diferentes filtros para que sejam aplicados à imagem. Um dos filtros disponibilizado chama-se *Cómico*, sendo este o mais próximo de um desenho. O resultado é apresentado na Figura 12.

Figura 12 – Caricatura gerada pelo aplicativo Clip2Comic



Fonte: Do Autor (2021)

Face Caricature - Cartoon Photo: pode-se realizar um *upload* de uma imagem da memória do celular ou tirar uma foto. No entanto, o aplicativo não gera uma caricatura, apenas disponibiliza imagens onde pode-se encaixar a foto original, em um corpo diferente, conforme exemplificado na Figura 13.

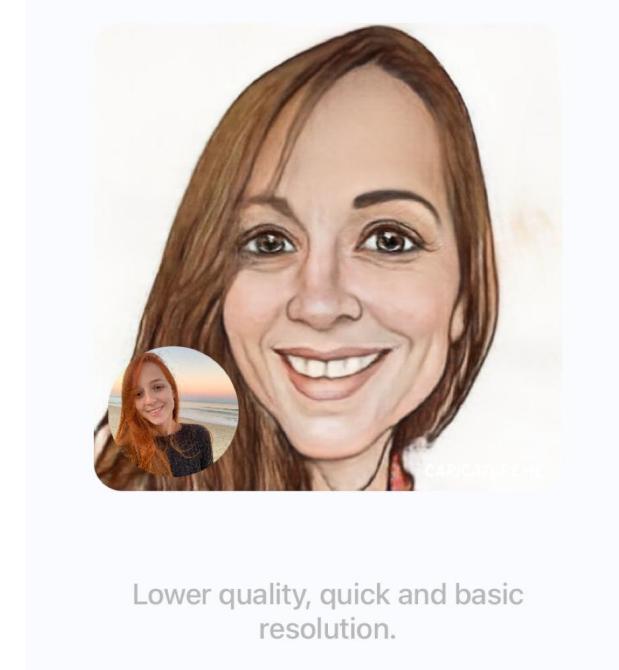
Figura 13 – Caricatura gerada pelo aplicativo Face Caricature - Cartoon Photo



Fonte: Do Autor (2021)

Caricature Me: também se realiza o *upload* de uma imagem da memória do celular e o aplicativo gera a caricatura com base nela. O aplicativo disponibiliza duas versões da caricatura, sendo uma delas paga. A Figura 14 apresenta a opção gratuita gerada com o aplicativo.

Figura 14 – Caricatura gerada pelo aplicativo Caricature Me



Fonte: Do Autor (2021)

Como pode-se observar, as caricaturas geradas por meio de aplicativos normal-

mente não realizam o exagero de alguma característica da pessoa retrada; apenas realizam a aplicação de algum tipo de filtro ou técnica para tornar a imagem base mais parecida com um desenho.

Entendendo os conceitos relevantes a construção de caricaturas, bem como as diferentes possibilidades de processo de construção da arte, no capítulo a seguir, será apresentada uma revisão sistemática realizada para entender quais são as técnicas de Criatividade Computacional que vêm sendo utilizadas na geração de caricaturas e como essas pesquisas têm lidado com as questões que caracterizam uma caricatura de sucesso.

5 REVISÃO SISTEMÁTICA

Para entender o que já vem sendo desenvolvido na área de geração automática e semiautomática de caricaturas, foi realizada uma revisão sistemática (KITCHENHAM; CHARTERS, 2007), buscando o estado da arte. Para tal, desenvolveu-se um protocolo contendo uma *string* de busca, etapas de leitura, critérios de inclusão e exclusão e perguntas norteadoras. Por fim, também foram definidas duas bases para a busca de artigos publicados, sendo elas *Web of Science* e a *IEEE Xplore*.

Os estudos deveriam conter obrigatoriamente em seu conteúdo o termo *artificial intelligence* ou *computational creativity*, bem como algum dos dois termos *caricature* ou *cartoon*, organizadas na seguinte *string* de busca: *ALL=((“artificial intelligence”OR “computational creativity”) AND (caricature OR cartoon))*. Ela foi aplicada às bases de dados e teve um retorno de 72 artigos na *Web of Science* e 81 artigos na IEEE, totalizando 153 trabalhos encontrados.

No protocolo inicial, foi definido que os estudos passariam por três etapas principais de leitura, os quais seriam classificados ou excluídos conforme os critérios. Na primeira etapa, foram lidos títulos, palavras-chave e resumo. Na segunda etapa, foram lidas a introdução e a conclusão. Por fim, os artigos remanescentes foram lidos em sua íntegra, respondendo às questões norteadoras.

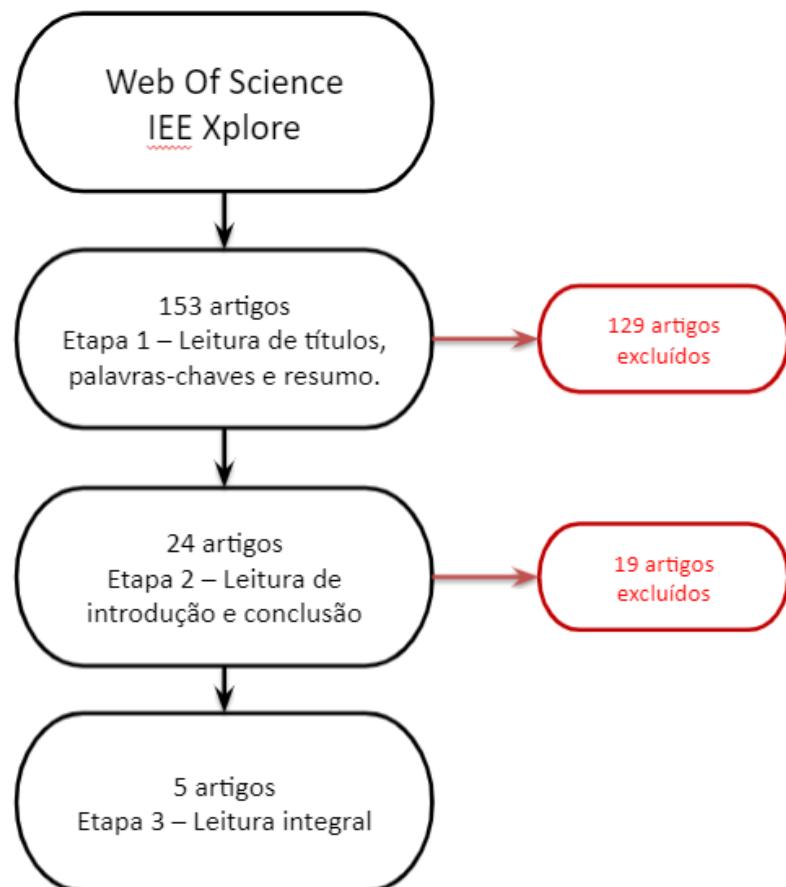
Foi realizada uma busca prévia em português, mas o retorno não foi significativo para a pesquisa. Por isso, optou-se por focar a revisão em estudos publicados na língua inglesa. Os critérios de inclusão (I) e exclusão (E) dos estudos são listados a seguir:

- (I) o artigo deve conter em suas palavras-chave e/ou resumo os termos *Criatividade Computacional* ou *Inteligência Artificial*, ou ainda alguma de suas técnicas;
- (I) o artigo deve estar publicado em inglês;
- (I) o artigo deve conter um desenvolvimento ou uso de ferramenta;
- (I) o artigo deve estar disponível de forma gratuita ou com acesso pela biblioteca da Universidade Feevale;
- (E) o artigo não está no contexto do presente estudo.

Tendo o retorno das bases de dados selecionadas, foi utilizado o *software State of the Art through Systematic Review (StArt)*, que é *open source*, para facilitar o processo de classificação dos artigos. O protocolo da revisão é cadastrado na ferramenta e os resultados das buscas são importados para ela, por meio de arquivos BibTex (HERNANDES *et al.*,

2010), gerados após a busca nas bases de dados. A partir disso, a classificação dos artigos é realizada de maneira organizada e prática, utilizando as funcionalidades disponíveis. A Figura 15 apresenta um fluxograma do resultado de cada fase de leitura, trazendo em vermelho o número de artigos excluídos em cada uma delas.

Figura 15 – Fluxograma de etapas de leitura com resultados



Fonte: Do autor (2021)

O retorno da aplicação da *string* de busca nas bases citadas foi de 153 artigos. Após uma primeira leitura, 129 artigos foram excluídos por não se encaixarem nos critérios definidos, restando 24 artigos. Destes, com a segunda etapa de leitura, foram excluídos 19 artigos, restando cinco artigos para leitura completa e que responderam às questões norteadoras deste estudo. Por meio dos artefatos encontrados nesta revisão, foi iniciado o processo de desenvolvimento para a solução do problema proposto, já que elas evidenciam as técnicas utilizadas, materiais e modelos de validação de resultados utilizadas pelos estudos.

A Tabela 1 apresenta a lista dos artigos selecionados na última etapa, juntamente com seu ano de publicação e autores. Elas estão organizadas na ordem crescente considerando o ano de publicação.

Tabela 1 – Artigos por ano de publicação, título e autor

Ano	Título	Autores
2008	<i>3D Caricature generation by manifold learning</i>	Pengfei Li , Yiqiang Chen , Junfa Liu , Guanhua Fu
2009	<i>Application of Facial Feature Localization Using Constrained Local Models in Template-Based Caricature Synthesis</i>	Lei Wei, Rui Mo, Wei Gao, Yi Zhu, Zhenyun Peng, Yaohui Zhang
2011	<i>Automatic Caricature Generation Using Text Based Input</i>	Kahkasha I. Siddavatam e Irfan A. Siddavatam
2020	<i>CaricatureShop: Personalized and Photorealistic Caricature Sketching</i>	Xiaoguang Han, Kangcheng Hou, Dong Du, Yuda Qiu, Yizhou Yu, Kun Zhou, Shuguang Cui
2020	<i>CariGAN: Caricature Generation through Weakly Paired Adversarial Learning</i>	Wenbin Li, Wei Xiong, Haofu Liao, Jing Huo, Yang Gao, Jiebo Luo

Fonte: Do autor (2021)

Ao todo, foram definidas quatro perguntas, que destacassem artefatos que foram utilizados, bem como processo de desenvolvimento e validação. São elas:

1. Quais técnicas foram utilizadas no estudo?
2. Que base de dados foi utilizada para treinamento e testes?
3. Como o trabalho foi validado?
4. Que características faciais foram utilizadas?

A seguir são detalhadas as respostas encontradas nos artigos selecionados para essas questões.

5.1 QUAIS AS TÉCNICAS UTILIZADAS NO ESTUDO?

Os artigos selecionados apresentaram diversas técnicas utilizadas para a geração de caricaturas. No trabalho apresentado por Li *et al.* (2008), são evidenciadas as técnicas *Locally Linear Embedding* (LLE) e *Extreme Learning Machine* (ELM) utilizadas no processo de caricaturas 3D baseadas em fotografias 2D.

Com relação às técnicas, Wei *et al.* (2009) descrevem um método de três etapas, no qual, primeiramente *Constrained Local Models* (CLM) e *Convex Quadratic Fitting* (CQF) são aplicadas para localizar características faciais chaves. Após isso, é realizada uma classificação dos componentes faciais em diferentes categorias baseadas na fisionomia Chinesa.

Por fim, modelos de componentes faciais são selecionados de acordo com a classificação realizada e montados em uma caricatura final.

O trabalho de Li *et al.* (2020), propõe uma nova técnica denominada *CariGan*, baseada em *Conditional Generative Adversarial Networks (GANs)* com treinamento fracamente pareado. Já o trabalho de Siddavatam e Siddavatam (2011) utilizou para a geração de caricaturas os algoritmos *Beier-Neely geometric warping* e *Mesh warping*.

5.2 QUE BASE DE DADOS FOI UTILIZADA PARA TREINAMENTO E TESTES?

Dos artigos selecionados, três utilizaram *datasets* de imagens e respectivas caricaturas. Um desses artigos, intitulado *3D Caricature generation by manifold learning*, apresentou um desenvolvimento de geração de caricaturas 3D a partir de fotos em 2D e, para isso, utilizou uma base de dados com duas partes: 110 fotografias reais em 2D e sua correspondência em 3D (LI *et al.*, 2008).

Em outro desses artigos, que abordou a utilização de *datasets*, foi relatada a construção de duas bases para treinamento e teste de Redes Neurais Profundas. O primeiro foi constituído de dados sintéticos, que mapeavam esboços parciais para um mapa denso de exagero. O segundo, por sua vez, foi formado por um conjunto de fotografias de alta resolução. De acordo com o estudo, ambos foram disponibilizados para uso de outros pesquisadores (HAN *et al.*, 2018).

Já o artigo CariGan aborda a utilização do conjunto de dados *WebCaricature*, disponível gratuitamente na internet. O *dataset* contém 5974 fotografias e 6042 imagens de caricaturas, de um total de 252 celebridades. Cada imagem possui cerca de 17 pontos de referência facial. Para o treinamento em questão, foram utilizadas 200 desse total e reservadas 52 para os testes (LI *et al.*, 2020).

O quarto estudo lido, não relatou a utilização de uma base de dados específica, mas mencionou a utilização de componentes faciais baseadas na fisionomia chinesa, juntamente com a respectiva caricatura do componente (WEI *et al.*, 2009). Por fim, o artigo que apresentava a geração de caricaturas utilizando entradas baseadas em texto não mencionou o conjunto de dados utilizados para treinamento ou testes (SIDDAVATAM; SIDDAVATAM, 2011).

5.3 COMO O TRABALHO FOI VALIDADO?

Em relação à validação, a grande maioria dos trabalhos descreveu a etapa aplicando seu método em algumas imagens não utilizadas nas fases de treinamento e apresentando os resultados gerados no artigo. Segundo Li *et al.* (2008), em seu estudo, foram escolhidas 20 fotografias para a geração de caricaturas, evidenciando, assim, os resultados das técnicas aplicadas.

Da mesma forma, os artigos de Wei *et al.* (2009) e Siddavatam e Siddavatam (2011) também apresentam alguns dos resultados gerados com a aplicação das técnicas descritas, mas não apresentam ou descrevem nenhuma forma de validação mais detalhada.

Já no trabalho de Han *et al.* (2018), foram selecionados aleatoriamente 10% dos dados utilizados como um conjunto de validação. Além disso, foram avaliados resultados qualitativos, selecionando 33 fotografias na internet e aplicando o processo desenvolvido. Os resultados dessa geração são apresentados ao longo do texto e analisados de forma qualitativa. Para validação sobre as questões de exagero no formato 3D, uma comparação entre o resultado gerado e outras duas técnicas, denominadas *naiveDeform* e *deepSketch2Face*, foi realizada. Por fim, o trabalho ainda apresenta uma comparação com outras duas técnicas, buscando uma validação qualitativa da síntese fotográfica. Para isso, as imagens foram geradas utilizando as técnicas *2DWarping* e *3D-Warping* e comparadas com os resultados do trabalho em questão.

Com uma validação mais robusta, Li *et al.* (2020) descrevem a validação da técnica desenvolvida por meio da comparação com outras técnicas e também um processo de entrevista com 16 participantes, na qual eles deveriam pontuar a cada imagem gerada de acordo com três aspectos: (1) plausibilidade; (2) preservação da identidade; e (3) exagero. Cada imagem recebeu uma pontuação entre 1 e 10 e, por isso, foi realizada a média das pontuações dos participantes.

5.4 QUE CARACTERÍSTICAS FACIAIS FORAM UTILIZADAS?

Os artigos lidos não descrevem, de forma clara, quais foram as características escolhidas ou descartados na realização do estudo. Por outro lado, a grande maioria utiliza técnicas para a extração dos pontos característicos da face (olhos, bocas, nariz, etc) e os compara com uma face média ou componentes faciais, gerando o exagero com base no resultado dessa comparação.

As respostas das questões descritas nortearam o desenvolvimento do protótipo proposto neste trabalho. Com base nas informações encontradas nessa revisão, foram definidas as tecnologias e técnicas a serem utilizadas na detecção de pontos principais da face ou na geração da caricatura em si. Além disso, a partir dessas respostas, foram identificados *datasets* para uso e formas de validação efetiva para os resultados gerados.

O capítulo seguinte apresenta o processo de desenvolvimento, o qual inclui os artefatos utilizados, bem como o processo de treinamento da rede utilizada.

6 DESENVOLVIMENTO

Neste capítulo são apresentadas as tecnologias e os materiais utilizados no desenvolvimento do protótipo, bem como seu processo de construção. Por meio das seções a seguir, é possível entender as etapas que constituíram e geraram os resultados finais.

6.1 BASE DE DADOS UTILIZADA

A base de dados utilizada para o treinamento foi a WebCaricature, encontrada por intermédio da revisão sistemática apresentada anteriormente e bastante utilizada em trabalhos que envolvem a geração de caricaturas ou identificação a partir delas (HUO *et al.*, 2018) (HUO *et al.*, 2017).

O *dataset* contém 6042 caricaturas e 5974 fotografias de 252 pessoas, coletadas da internet. Além disso, para cada imagem, são disponibilizados 17 marcos faciais, os quais não são utilizados neste trabalho. As imagens não tem nenhum tipo de padrão de iluminação, pose ou expressão, e as caricaturas são apresentadas em diferentes estilos artísticos.

A Figura 16 apresenta alguns exemplos de fotos e suas respectivas caricaturas encontradas na base de dados em questão.

Figura 16 – Exemplos de imagens da base de dados



Fonte: Huo *et al.* (2018)

Para cada uma das pessoas apresentadas na primeira coluna, são exibidas, nas colunas seguintes, seis diferentes caricaturas. Por meio dos exemplos, pode-se perceber que os dados não são pareados, ou seja, apesar de se referirem a mesma pessoa, caricaturas e fotos não estão organizadas em mapeamento um-para-um nos domínios de origem e destino, sendo independentes e expostas a condições diferentes. Não foram encontradas bases de dados com uma configuração diferente; assim, optou-se pela utilização da base em questão.

Apesar de o *dataset* ser gratuito para a utilização, é necessário solicitar permissão de acesso aos seus idealizadores. Para isso, é disponibilizado, no site, um termo de responsabilidade, com algumas questões a serem respondidas, que identificam o motivo da solicitação de uso, o qual deve ser enviado via e-mail aos pesquisadores. O formulário enviado encontra-se no Apêndice A deste trabalho. Como retorno, o acesso à base é liberado por meio de um *link* do *Google Drive*. O e-mail de autorização encontra-se no Apêndice B deste trabalho. Os dados são zipados e organizados pelo nome da pessoa apresentada; por isso, necessitam de uma reorganização, apresentada na seção 6.5, que retrata o processo de desenvolvimento.

Na próxima seção, são apresentadas, de forma breve, as tecnologias utilizadas no desenvolvimento deste trabalho.

6.2 REDES ADVERSÁRIAS GENERATIVAS (GANS)

As Redes Adversárias Generativas (GANs) foram apresentadas em 2014 por Ian Goodfellow e outros pesquisadores da universidade de Montreal (GOODFELLOW *et al.*, 2014). As GANs são classificadas como modelos generativos, por serem capazes de gerar novos artefatos, sejam eles imagens, músicas, poesias, dentre outros.

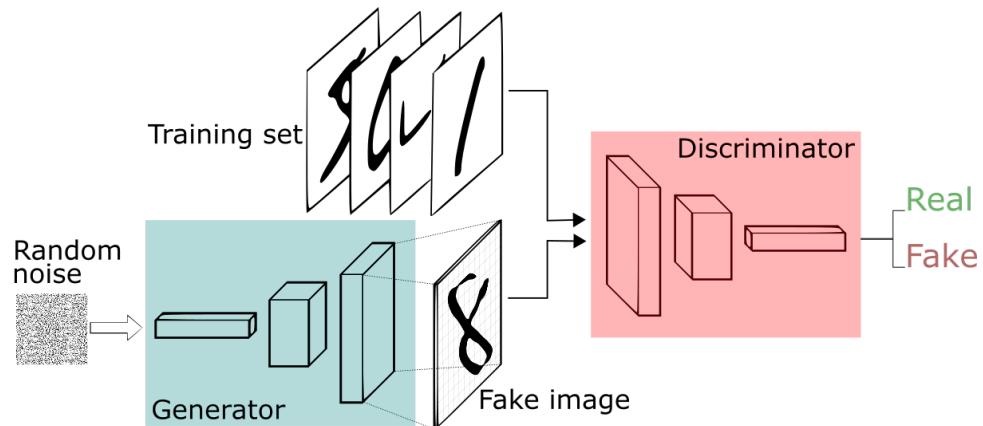
Elas são constituídas por redes neurais profundas, com dois componentes principais: um discriminador e um gerador. O discriminador é uma rede classificatória, treinada com dados reais e dados sintéticos, para diferenciar o conteúdo real do conteúdo falsificado. O gerador, por sua vez, recebe um vetor de ruído aleatório e é treinado para gerar dados sintéticos. Ambos vão aprimorando seus métodos: o discriminador, buscando maximizar sua capacidade de diferenciar os dados e o gerador, tentando diminuir a diferença entre os dados que gera para os dados reais. Assim, a ligação entre os componentes encontrados na rede é dita adversária, já que as duas redes irão competir entre si (CRESWELL *et al.*, 2018).

O discriminador é simplesmente um classificador binário e é descartado após o treinamento (GOODFELLOW; BENGIO; COURVILLE, 2016). O treinamento do gerador é um pouco mais complicado e envolve as duas redes. Após o gerador realizar a classificação, propaga-se de volta a perda de classificação binária, por meio do discriminador e do

gerador e, com base nisso, são atualizados apenas os pesos do gerador.

A Figura 17 apresenta, de forma gráfica, como é organizada uma GAN, na qual o gerador recebe como entrada um vetor de valores randômicos, gerando uma imagem. Por sua vez, o discriminador recebe imagens geradas pelo gerador e também imagens reais, classificando-as como verdadeiras ou falsas.

Figura 17 – Exemplo de arquitetura de modelo de rede adversária generativa



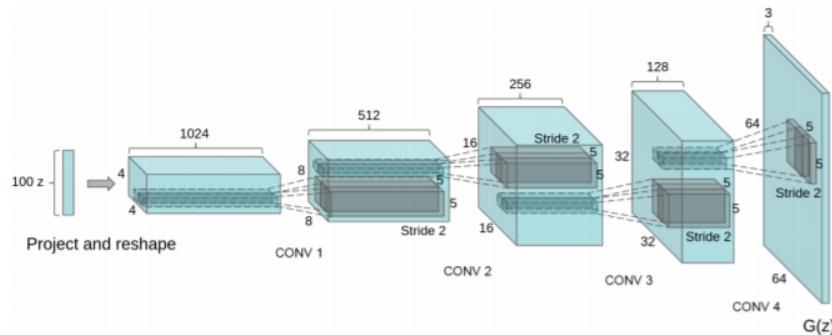
Fonte: Brownlee (2019)

Existem diferentes variações dessa arquitetura, como, por exemplo, a GAN convolucional profunda (DCGAN) e a CycleGAN, também utilizadas neste trabalho. Na seção a seguir, explica-se a estrutura de DCGAN.

6.3 DCGAN

Esta arquitetura de GAN foi proposta por Radford, Metz e Chintala (2015) e, conforme seu nome sugere, utiliza *Convolutional Neural Network* (CNN) para a estruturação do gerador e do discriminador. A Figura 18 apresenta sua estrutura.

Figura 18 – Estrutura de DCGAN



Fonte: Radford, Metz e Chintala (2015)

De acordo com Radford, Metz e Chintala (2015), que propõe a arquitetura em questão, a DCGAN substitui as camadas de *pooling*, em que são recebidas cada saída do mapa de características da camada de convolução. Após isso, ele prepara um mapa de características condensadas, por camadas de *strided convolution*, na qual é realizado um filtro na imagem conforme um intervalo pré-definido (ACADEMY, 2021). Além disso, utiliza *batch normalization* no discriminador e no gerador, que é um processo para tornar as redes neurais mais rápidas e estáveis, utilizando camadas extras, que executam as operações de padronização e normalização na entrada de uma camada recebida da camada anterior. Também, ele faz o uso de *LeakyRelu* como função de ativação do discriminador (RADFORD; METZ; CHINTALA, 2015).

Outra arquitetura GAN utilizada neste trabalho, é apresentada na próxima seção.

6.4 CYCLEGAN

Dentre as diferentes arquiteturas de redes geradoras, existe a CycleGAN, uma técnica GAN que é composta por dois geradores e dois discriminadores, formando a ideia de ciclo. Assim, os dois modelos fazem a tradução de imagem a imagem, mas em direções opostas. O primeiro gerador, denominado G, converte as imagens do domínio A para o domínio B. O segundo gerador, denominado F, converte as imagens do domínio B para o A. Cada um dos geradores tem seu respectivo discriminador *PatchGAN*, que tenta diferenciar as imagens geradas das imagens reais.

Os modelos são treinados de forma não supervisionada, com a utilização de dados do domínio de origem e do domínio de destino, que não necessitam estar relacionados. Alguns exemplos desse tipo de cenário são: a tradução de uma paisagem de verão em inverno ou a tradução de pinturas em fotografias.

Quanto à estrutura do gerador da CycleGAN, Bansal e Rathore (2017) afirmam que, em geral, ela consiste em um modelo com três componentes principais. São elas:

1. Codificador
2. Transformador
3. Decodificador

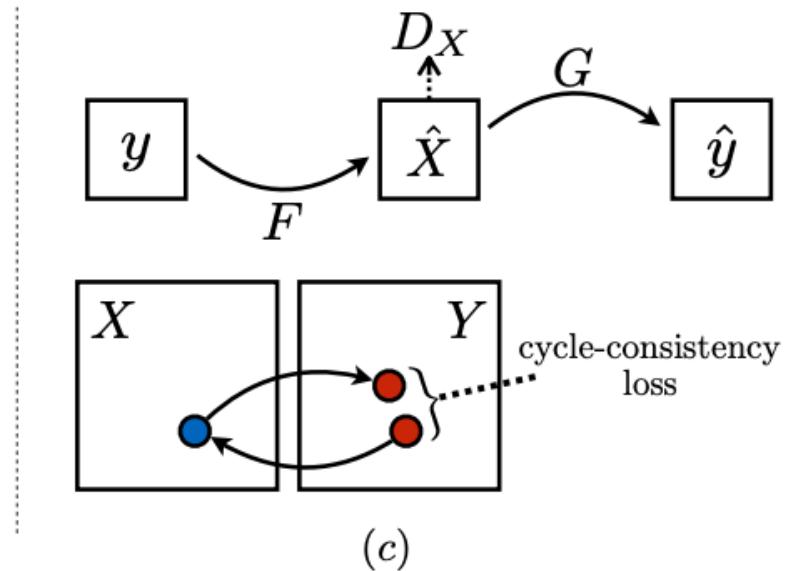
Na primeira etapa, de codificação, o recurso da imagem é extraído por meio de uma rede de convolução (UJJWALKARN, 2016). O codificador tem três convoluções que reduzem a imagem em 1/4 do seu tamanho real. Na etapa de transformação, a entrada será a saída do codificador, após a aplicação de ativação. Nela, há uma combinação de diferentes recursos para transformar o vetor, mantendo algumas características da imagem

original, utilizando blocos residuais. Por fim, a etapa de decodificação fará o oposto da etapa 1, realizando a deconvolução (BANSAL; RATHORE, 2017).

A grande vantagem da utilização de um modelo cycleGAN é a não necessidade da utilização de amostras pareadas. No cenário deste trabalho, outra vantagem é que sua entrada não será um vetor aleatório. Como os dados não são pareados, um dos grandes problemas, segundo Zhu *et al.* (2017), seria a possibilidade de uma geração final completamente aleatória no domínio de destino, não condizente com o esperado. Para solucionar esse problema, a arquitetura de CycleGAN trabalha com a perda de consistência do ciclo. Essa perda se baseia na ideia de que se converter uma imagem de um domínio para o outro e, em seguida, realizar o processo reverso; o resultado deve estar próximo da imagem inicial. A diferença entre as duas imagens é chamada de perda de consistência de ciclo.

A Figura 19 demonstra o processo desse tipo de perda. Nesse caso, pega-se a imagem X e é gerada Y. Em seguida, convertida a imagem Y é convertida para o domínio X. A imagem X gerada deve ser quase idêntica à imagem original. Essa perda será utilizada para atualizar o modelo, como parte da função de perda final.

Figura 19 – Exemplo de arquitetura de modelo de rede adversária generativa



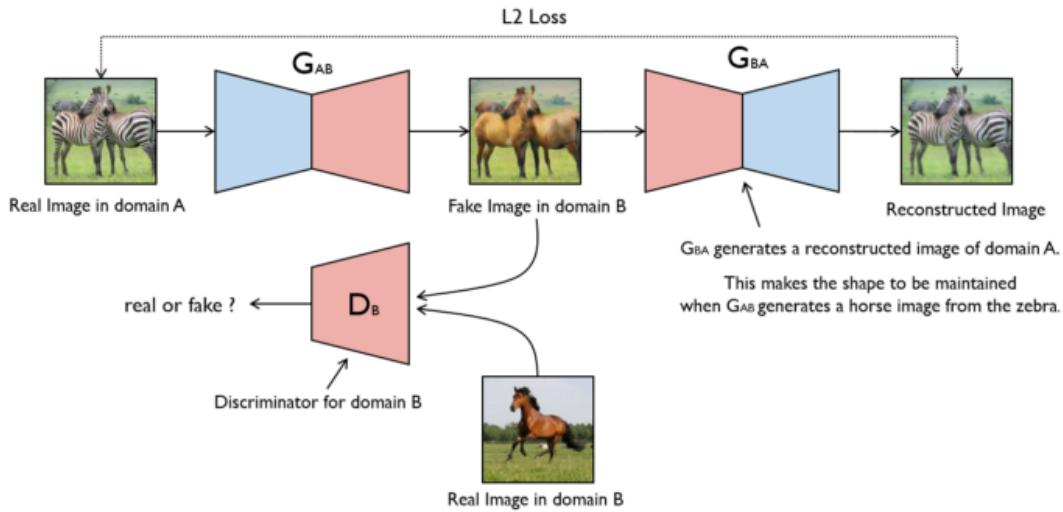
Fonte: Zhu *et al.* (2017)

Um exemplo muito utilizado é a ideia de que se alguém traduzir uma frase em inglês para o francês, e depois traduzi-la de volta do francês para o inglês, a frase deve ser igual a frase original (ZHU *et al.*, 2017).

Outro exemplo muito conhecido dessa arquitetura é a ideia de transformar uma zebra em cavalo. Não há dados reais pareados deste tipo de situação; então, é possível observar, com clareza o comportamento da estrutura. A Figura 20 apresenta a arquitetura da *CycleGan*, representada com o modelo da tradução da zebra para o cavalo, construída

pelos próprios autores da arquitetura em questão.

Figura 20 – Arquitetura do modelo CycleGAN



Fonte: Zhu *et al.* (2017)

Nela, tem-se uma imagem real de zebras, que são transformadas pelo primeiro gerador (G_{AB}) em cavalos. A imagem em questão será avaliada pelo discriminador, que recebe a imagem gerada e também imagens reais de cavalos. O segundo gerador (G_{BA}) também receberá a imagem de cavalos gerada e tentará transformá-los em zebras novamente. A zebra gerada pelo segundo gerador (G_{BA}) será comparada com a imagem de entrada do primeiro gerador (G_{AB}), calculando a perda de consistência de ciclo. Além disso, o segundo gerador também tem seu próprio discriminador, que realizará a avaliação da zebra gerada, mas que não está representado na figura em questão. Ao final, as perdas dos discriminadores são somadas à perda de consistência de ciclo e propagadas para a atualização dos pesos dos geradores.

Na próxima seção, são apresentadas as etapas do treinamento da rede neural, utilizando as diferentes arquiteturas mencionadas.

6.5 TREINAMENTO

Todas as tecnologias e materiais utilizados nesta etapa do trabalho foram definidas a partir dos estudos apresentados no Capítulo 5, com as respostas encontradas nas perguntas norteadoras. Assim, destaca-se a importância que a realização da revisão sistemática teve no presente trabalho.

Quanto ao treinamento, primeiramente, foi realizado um estudo prático utilizando uma estrutura de GAN simples, frequentemente empregada para introdução de GAN's. O código foi baseado no tutorial apresentado por Brownlee (2019). O ambiente utilizado para o treinamento foi o Google Colab, no qual foi possível gerar dígitos manuscritos,

conforme indicava o tutorial. A ideia inicial foi entender como a arquitetura funcionava e quais eram os tipos de resultados gerados a partir do treinamento realizado. Com isso, foi possível entender o funcionamento e comportamento de uma GAN.

Após esse estudo, o treinamento foi realizado com a utilização de um código DCGAN, que foi proposto por Radford, Metz e Chintala (2015). Essa arquitetura foi escolhida, porque é bastante utilizada para a geração de novas faces. Desse modo, ela poderia ser útil nesse contexto.

O código utilizado é disponibilizado pelo próprio Keras, encontrado na página do Keras¹, e não havendo necessidade de alteração. O código original utiliza o *dataset* celebA, que é disponibilizado por Liu *et al.* (2015), de forma gratuita, para uso não comercial. Para entender o funcionamento da arquitetura e seu retorno, foi realizado o treinamento inicial sem modificar os dados utilizados. Todavia, ao tentar rodar no Google Colab as 100 épocas, como a arquitetura anterior, o treinamento era parado pelo ambiente, indicando que não havia GPUS disponíveis e que era preciso contratar um plano pago que disponibilizaria mais limite de utilização das GPUs ou rodar em um ambiente sem a GPU. A empresa não explicita quais são os limites de uso para cada plano.

Com isso, foi decidido realizar o treinamento em um computador da Universidade Feevale que tem 2 GPUS RTX 2080TI. Para isso, foi realizado contato com o professor Gabriel da Silva Simões, que disponibilizou acesso e auxiliou no processo de treinamento.

Alguns dos resultados do treinamento são apresentados no compilado de imagens na Figura 21. Como é possível perceber, a rede conseguiu gerar novos rostos com as 100 épocas com as quais foi rodado.

Figura 21 – Resultado do treinamento da arquitetura DCGAN



Fonte: Do autor (2021)

Todavia, no *dataset* WebCaricature, os dados de fotos e as caricaturas não são pareados; além disso, na estrutura de DCGAN, não foi possível controlar a entrada de forma a limitar os domínios de saída da rede. Dessa forma, em algum ponto do treinamento, a estrutura utilizada pode começar a gerar imagens de caricatura aleatórias, que têm pouca ou nenhuma correlação com as faces que foram utilizadas inicialmente.

¹ https://keras.io/examples/generative/dcgan_overriding_train_step/

Realizada essa análise, e com base nos resultados da revisão sistemática, buscou-se experimentar outro tipo de arquitetura GAN, que possibilitasse mais controle dos resultados. A CycleGAN apresentou-se mais aderente à proposta. O código utilizado foi desenvolvido por Nain (2020) e disponibilizado pelo Keras para utilização gratuita². Não foram realizadas modificações na estrutura proposta, mas foi necessária uma adaptação no formato das imagens da base Webcaricature, para que fosse possível utilizá-las como entrada. A base de dados inicial era carregada pelo TensorFlow, com o formato *TFRecord*, que é o formato binário do próprio Tensorflow.

Após o carregamento dos caminhos de cada imagem em um *array*, que é aprofundado a seguir, buscou-se tornar os dados próximos ao formato que o código original esperava, por meio das funções em Python apresentadas na Figura 22

Figura 22 – Funções para adequação do formato dos dados

```
def parse_function(filename, label):
    image_string = tf.io.read_file(filename)
    image = tf.image.decode_jpeg(image_string, channels=3)
    image = tf.image.convert_image_dtype(image, tf.float32)
    resized_image = tf.image.resize(image, [64, 64])
    return resized_image, label

#codigo original do CycleGAN
def preprocess_train_image(img, label):
    # Random flip (rotacao)
    img = tf.image.random_flip_left_right(img)
    # Resize to the original size first
    img = tf.image.resize(img, [*orig_img_size])
    # Random crop to 256X256
    img = tf.image.random_crop(img, size=[*input_img_size])
    # Normalize the pixel values in the range [-1, 1]
    # img = normalize_img(img)
    return img

train_caricatures = tf.data.Dataset.from_tensor_slices((train_path_caricatures, train_path_caricatures))
train_caricatures = train_caricatures.map(parse_function, num_parallel_calls=autotune)
train_caricatures = train_caricatures.map(preprocess_train_image, num_parallel_calls=autotune)
train_caricatures = train_caricatures.cache()
train_caricatures = train_caricatures.shuffle(buffer_size)
train_caricatures = train_caricatures.batch(batch_size)
```

Fonte: Do autor (2021)

Os dados seriam transformados em um objeto combinando rótulos e imagens a partir da função do TensorFlow, *from_tensor_slices*. Após isso, passariam pela função *parse_function*, a qual carregaria as imagens e faria seu redimensionamento. Por fim, passariam pelo função do código original do CycleGAN, responsável pelo pré-processamento das imagens.

Apesar desse código funcionar, a solução adotada foi outra, visando à melhor performance. O processo realizado foi: (i) carregamento do *dataset* de zebras e cavalos do código original; (ii) encontrar o arquivo zip dentro da pasta de *download* do Tensorflow *dataset*; (iii) descompactar a pasta; (iv) trocar os dados para as imagens de caricaturas e fotos, respeitando a distribuição de treinamento e teste; e (v) zipar a pasta utilizando o mesmo nome inicial. Após isso, é necessário apagar a pasta *extracted* e a pasta que contém

² <https://keras.io/examples/generative/cyclegan/>

os arquivos em TFRecords. O Tensorflow não realiza o *download* dos dados sempre que é rodado, apenas se eles não existirem na máquina em que está sendo chamado. Assim, ele realizará a modificação dos dados em cima do novo zip, modificado para ter as novas imagens e os dados já chegarão ao código no formato ideal, de forma muito mais rápida.

Diferente de outras bases propostas para esse tipo de trabalho, que já são divididas em treinamento e teste, a WebCaricature é organizada em pastas por pessoa, ou seja, cada pessoa encontrada no *dataset* tem um diretório com seu respectivo nome, suas fotos e caricaturas. Para a organização desses dados, foi criado um índice, com o nome de cada uma das celebridades encontradas. Depois disso, utilizou-se um *script* em Python para carregar os caminhos de cada imagem em um array, utilizando um prefixo e adicionando o índice criado. Dessa forma, ao final da rotina, o array continha o caminho de todas as imagens. Além disso, as imagens e as caricaturas eram diferenciadas apenas pela primeira letra do nome, sendo iniciado 'P' de foto (*photo*) e 'C' de caricatura (*caricature*). Uma função lambda foi construída para classificar a imagem em foto ou caricatura, por meio do nome da imagem.

Com essas estruturas prontas, os caminhos foram classificados em fotos e caricaturas, sendo divididas em dois *arrays* diferentes. Após isso, cada um dos grupos foi embaralhado e dividido em outros dois grupos: um de imagens para teste, com cerca de 20 por cento dos dados, e outro para imagens de treinamento, com 80 por cento, baseado no tutorial³. Esses dados foram salvos no GoogleDrive com a divisão em questão e utilizados no modelo CycleGAN.

Para treinar o modelo, os dados no conjunto de treinamento são divididos em lotes, denominados como *batch_size*, inteirando repetidamente em todo o conjunto de dados para um conjunto de épocas definidas. Nesse caso, foram realizados dois treinamentos de 100 épocas com *batch_size* (tamanho de lote) diferentes, sendo eles no valor de 1 e 4. O hiperparâmetro citado é um termo utilizado no aprendizado de máquina em geral, e faz referência ao número de exemplos utilizados em uma iteração no treinamento. (ACADEMY, 2021). Também foi realizado um treinamento de 150 épocas com *batch_size* de 1, buscando compreender se o aumento de épocas e mudanças no parâmetro citado beneficiaria ao resultado final. Além disso, um quarto treinamento foi realizado, após um pré-processamento das imagens, no qual foram retirados os detalhes de seu fundo.

A retirada do fundo das imagens, mencionada anteriormente, foi realizada com o uso do modelo DeepLab V3, uma arquitetura semântica que lida com problemas de segmentação de objetos de múltiplas escalas, proposta pela Google (CHEN *et al.*, 2017). As funções utilizadas estão disponíveis no site da *LearnOpenCV*⁴ e foram incluídas no

³ <https://cs230.stanford.edu/blog/split/>

⁴ <https://learnopencv.com/applications-of-foreground-background-separation-with-semantic-segmentation/>

loop de carregamento das imagens.

A Figura 23 apresenta um exemplo de imagem antes e depois do processo de retirada dos detalhes de fundo. Com o tratamento, a imagem acaba por perder qualidade também.

Figura 23 – Exemplo de antes e depois do processo de retirada dos detalhes de fundo



Fonte: Do autor (2021)

A cada época, uma imagem de caricatura era salva para que fosse possível analisar os resultados gerados no decorrer do treinamento e também um *checkpoint* do modelo treinado. Cada época rodou em cerca de 10 minutos, somando um tempo de treinamento entre 15 e 20 horas para serem finalizados. Nas tabelas encontradas nos Apêndices C, D, E e F deste trabalho, são apresentados os resultados da função de perda (*loss*) por época de cada treinamento. Esses dados são importantes para entender em que época o modelo teve um valor de *loss* maior, indicando maior aprendizagem do modelo.

Na próxima seção, são apresentados os resultados dessas gerações mencionadas, bem como uma discussão a seu respeito.

6.6 RESULTADOS

Nesta seção, são apresentados os resultados oriundos dos treinamentos descritos na seção anterior. As imagens apresentadas em cada treinamento são baseadas na última época e na época que apresentou menor *loss* no gerador G, responsável por realizar a transformação da foto em caricatura durante o treinamento.

6.6.1 Treinamento de 100 épocas

O primeiro treinamento realizado foi configurado para rodar 100 épocas, sem alterações nos parâmetros configurados no código original, ou seja, tendo um *bath_size* com valor 1. A Figura 24 apresenta uma amostra de imagens aleatórias que foram geradas durante esse treinamento, em que, na primeira e na segunda coluna, são apresentadas imagens da época de número 100 e na última coluna, apresenta-se a época de número 38, devido a ela ter apresentado menor valor de *loss*.

Figura 24 – Resultados aleatórios gerados com treinamento de 100 épocas



Fonte: Do autor (2021)

As imagens demonstradas são do conjunto separado para testes. Todavia, após o treinamento, também foram tiradas fotos de diferentes pessoas em diferentes cenários para a realização de um teste com os modelos já treinados. Não foram controlados parâmetros como exposição de luz, expressões e fundos. As fotos foram autorizadas pelas pessoas envolvidas. Os resultados são apresentados na Figura 25, na qual a primeira coluna apresenta a foto que foi utilizada como entrada para o modelo; já segunda coluna apresenta o resultado gerado pelo modelo treinado com 100 épocas; por sua vez e na terceira coluna, os resultados gerados pelo modelo com 38 épocas.

Figura 25 – Utilização do modelo de 100 épocas



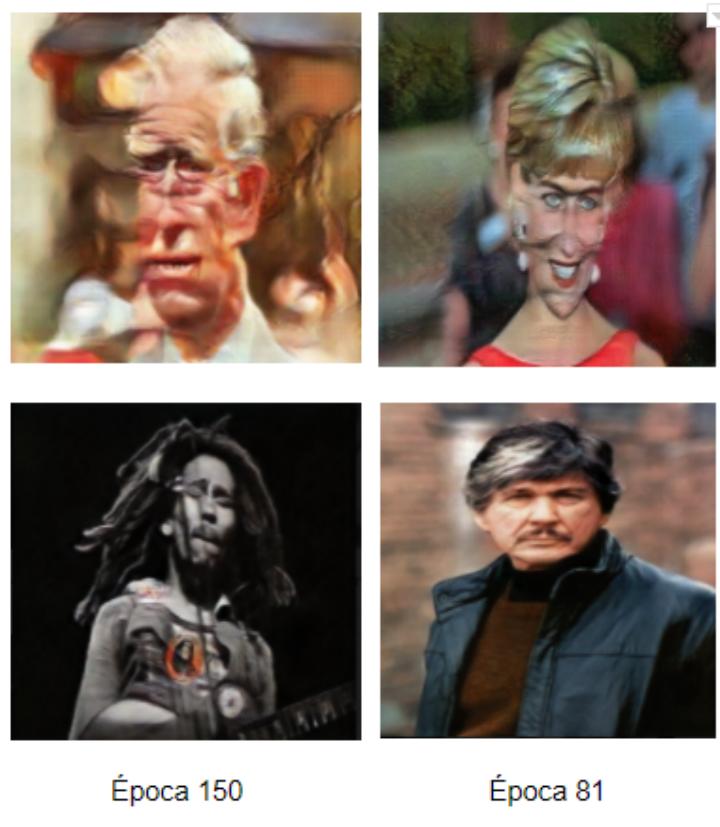
Fonte: Do autor (2021)

Considerando a imagem apresentada, é possível perceber que as imagens geradas pelo modelo com 38 épocas apresentam mais clareza, com menos manchas. Todavia, também não apresenta muitos exageros, importantes para uma caricatura. Outro fator a ser destacado é a imagem apresentada na terceira linha, que tem tons de laranja muito parecidos entre o fundo e o cabelo da modelo. Essa imagem apresentou uma qualidade inferior às demais, possivelmente por ter tons próximos.

6.6.2 Treinamento de 100 épocas com *batch_size* 4

Um novo treinamento foi rodado com a alteração do parâmetro *batch_size* para o valor de 4. Imagens aleatórias foram selecionadas na época 100 e na época 81, esta última por ter apresentado menor valor de *loss*. As imagens selecionadas são encontradas na Figura 26. O aumento do hiperparâmetro não alterou o tempo de treinamento, sendo este finalizado após cerca de 16 horas.

Figura 26 – Treinamento com 100 épocas e com hiperparâmetro *batch_size* 4



Fonte: Do autor (2021)

Apesar de o treinamento ter alcançado seu menor valor de perda em épocas maiores que o treinamento anterior, os resultados aleatórios, que foram gerados no treinamento, apresentam menor aproximação à ideia de caricatura. A Figura 27 apresenta um gráfico de comparação entre os resultados de *Loss* por época dos treinamentos descritos, tendo *batch_size* diferentes.

Figura 27 – Comparação de *Loss* entre treinamentos de batch_size diferentes por época



Fonte: Do autor (2021)

O modelo em questão também foi aplicado em fotos de pessoas. Os resultados são apresentados na Figura 28. A primeira coluna apresenta a imagem de entrada para o modelo; já segunda coluna apresenta o resultado do modelo treinado com 100 épocas; por sua vez, na terceira, imagens geradas com o modelo salvo no *checkpoint* de 81 épocas.

Figura 28 – Utilização do modelo de 100 épocas com batch_size 4



Fonte: Do autor (2021)

As imagens geradas com o modelo de melhor resultado de *loss* apresentam clareza na pessoa representada; todavia, não apresentam exageros ou modificações significativas para representar uma caricatura. Pode-se perceber que os modelos gerados nesse trei-

namento apresentaram melhor adaptação com imagens de tons parecidos entre pessoa e fundo, quando comparados com o modelo anterior.

6.6.3 Treinamento de 150 épocas

Após os treinamentos anteriores, foi decidido aumentar o número de épocas, para realizar uma análise do comportamento do modelo nestas condições. Os demais hiperparâmetros foram mantidos do código original, ou seja, o *batch_size* retornado para 1. As imagens aleatórias geradas no decorrer do treinamento são apresentadas na Figura 29, na qual as duas linhas se referem ao mesmo treinamento, mas a primeira coluna apresenta os resultados da época 150, e a segunda coluna representa a época 144, considerada a melhor de acordo com a função de perda.

Figura 29 – Resultados gerados durante treinamento com 150 épocas

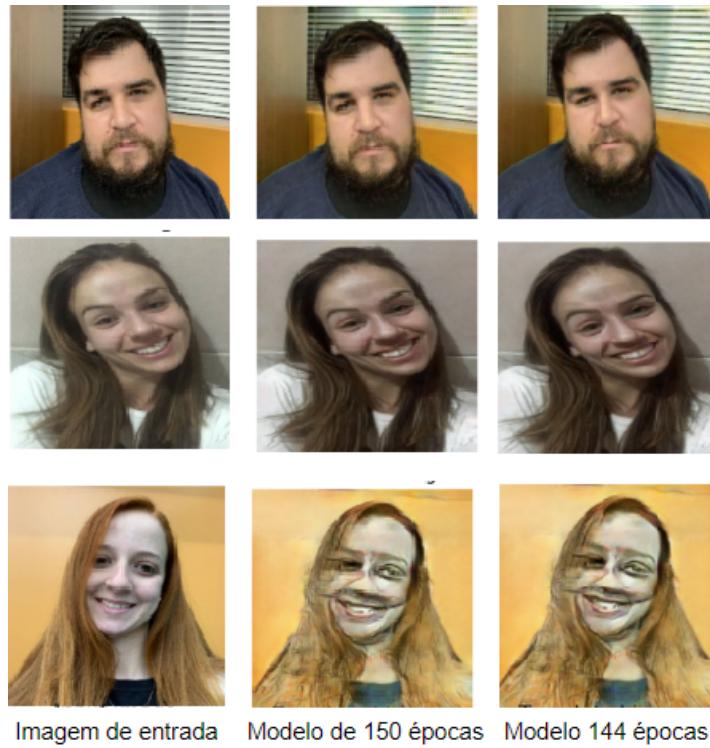


Fonte: Do autor (2021)

Destaca-se que, nas duas épocas, a rede apresentou imagens de boa resolução e também imagens borradadas, como vistas na segunda linha. Além disso, conforme realizado nos outros dois treinamentos, o modelo final e o com menor resultado de *loss* foi aplicado em fotos e apresentadas na Figura 30, a seguir.

Conforme visto na imagem, o modelo de 150 épocas não teve grandes diferenças dos resultados apresentados nos treinamentos anteriores, trazendo as mesmas questões

Figura 30 – Aplicação do modelo de 150 épocas



Fonte: Do autor (2021)

em relação à falta de exagero nas imagens e à forma como lida com cores próximas entre fundo e pessoa.

6.6.4 Treinamento com imagens com fundo neutro

Conforme mencionado na seção anterior, as imagens do último treinamento receberam um pré-processamento, tornando o fundo das imagens neutro. Isso foi realizado buscando entender se os detalhes de fundo das imagens afetavam a capacidade do modelo em aprender a modificar o rosto da pessoa apresentada. Ao analisar as imagens geradas no decorrer do treinamento, grande parte das imagens com detalhes de fundo apresentavam qualidade inferior e, por vezes, traziam borrões.

Assim, o treinamento foi realizado com 100 épocas e *batch_size* com valor 1. A Figura 31 apresenta as imagens aleatórias geradas no período de treinamento do modelo, com 100 épocas e com 41 épocas, considerada a melhor de acordo com o *loss*.

Vale ressaltar que o processo de retirar os detalhes de fundo das imagens em grande quantidade também pode ter diminuído a sua qualidade. Apesar de as imagens de teste não parecerem muito promissoras quanto à aproximação às caricaturas, ao realizar a aplicação do modelo nas fotos, o resultado demonstrou-se satisfatório. As imagens em questão são apresentadas na Figura 32, na qual, assim como as demais, a primeira coluna representa a imagem de entrada utilizada e as próximas são imagens de saída da rede.

Figura 31 – Imagens geradas pelo treinamento usando imagens com fundo neutro



Fonte: Do autor (2021)

Figura 32 – Aplicação do modelo treinado com imagens pré processadas



Fonte: Do autor (2021)

As imagens apresentam alterações interessantes, que podem representar uma caricatura, devido a elas manterem a identidade da pessoa representada e a fazerem algumas alterações. A primeira imagem utilizada como *input* não apresentou resultado satisfatório nesse experimento, apesar de ter sido uma das melhores nos cenários anteriores.

Apresentadas as caricaturas desenvolvidas nesta pesquisa, no próximo capítulo, apresenta-se a conclusão deste estudo, bem como possíveis oportunidades de melhorias e

investigações futuras.

7 CONCLUSÃO

Com os estudos realizados no presente trabalho, buscou-se contextualizar os conceitos e as práticas da Criatividade Computacional. O desenvolvimento da revisão sistemática, apresentado no Capítulo 5, salientou algumas das técnicas que estão sendo utilizadas na geração de caricaturas na área de Criatividade Computacional, bem como evidenciou o processo de desenvolvimento, os *datasets* utilizados e as formas de validação utilizadas. Essas informações foram úteis nas etapas de desenvolvimento deste trabalho, nas quais foram utilizados os resultados coletados para definir tecnologias e materiais usados para a execução dessa investigação.

Para o desenvolvimento deste trabalho, também foi necessário compreender conceitos de criatividade e, especificamente, da Criatividade Computacional. Conforme apresentado, desde o capítulo introdutório, a criatividade é um termo que compreende diferentes aspectos e, desse modo, é importante definir um conceito a ser seguido ao decorrer do presente trabalho (PIRES, 2017). Quanto ao campo da Criatividade Computacional, buscou-se entender não somente seu significado e técnicas, mas também um pouco de sua evolução histórica e de algumas de suas aplicações.

Outro campo estudado foi o da Caricatura. Compreender sua história e as técnicas utilizadas serviram como base para a aplicação das técnicas de Criatividade Computacional, bem como do melhor entendimento dos resultados gerados. Nesse sentido, buscou-se entender alguns modelos de criação, além de perceber, por meio da análise de caricaturas já prontas, as características das faces e como elas foram exageradas. Além disso, aplicativos gratuitos, que geram caricaturas a partir de fotos, foram testados. Nestes percebe-se pouca ou nenhuma distorção das características principais, fugindo da essência da técnica de caricatura ou, ainda, apresentando imagens distorcidas.

Ao realizar o treinamento da rede neural, percebeu-se que a arquitetura GAN é bastante aderente a esse tipo de proposta, mas que utilizando as arquiteturas atuais, sem realizar alterações, não é possível ter um maior controle do que será mudado nas fotos de entrada. A rede decide por si o que será alterado. Dessa forma, não há como garantir que são realmente as características principais da pessoa retratada que estão sendo exageradas. Por outro lado, apesar das alterações realizadas pela rede, a identidade é mantida, tornando possível reconhecer a pessoa que está sendo caricaturada. Esses dois fatores foram apresentados como importantes na arte da caricatura.

Outro objetivo foi realizado por meio da revisão sistemática, a qual foi apresentada no Capítulo 5, no qual, através da literatura, foi possível conhecer as técnicas já utilizadas para tal objetivo. A revisão também possibilitou selecionar o *dataset* utilizado. Por fim,

o protótipo foi desenvolvido e seus resultados discutidos. Com isso, pode-se afirmar que os objetivos específicos também foram alcançados.

Como trabalho futuro, pode-se buscar alguma forma de treinar a rede para identificação desses pontos de características principais da face, com a finalidade de gerar alterações em cima delas. Para isso, é necessário um aprofundamento no estudo da arquitetura para realizar alguma alteração, buscando alcançar esse resultado. Ademais, outra oportunidade encontrada é o desenvolvimento de uma rede para avaliar as caricaturas geradas, com o objetivo de retirar a subjetividade da avaliação humana.

É interessante destacar, conforme diz Colton, Wiggins *et al.* (2012), que a Criatividade Computacional utiliza técnicas de Inteligência Artificial, buscando criar novos artefatos. Nesse ínterim, este trabalho buscou experimentar algumas técnicas e entender os resultados gerados. Por essa razão, existem diferentes técnicas e modificações nas arquiteturas atuais que podem ser testadas para esse fim.

REFERÊNCIAS

- ACADEMY, D. S. *Deep Learning*. Book Brasil, 2021. Disponível em: <<https://www.deeplearningbook.com.br/>>. Acesso em: 22 out. 2021. Citado 2 vezes nas páginas 36 e 41.
- AMABILE, T. M. *Managing for creativity*. [S.l.]: Harvard Business School Background Note, 1996. 396–271 p. Citado 2 vezes nas páginas 14 e 15.
- AMABILE, T. M. *How to kill creativity*. [S.l.]: Harvard Business School Publishing Boston, MA, 1998. Citado na página 14.
- AMARILDOCHARGE. *BLOG DO AMARILDO. CHARGE CARICATURA*. 2011. Disponível em: <<https://amarildocharge.wordpress.com/2011/10/27/dilma-caricatura/>>. Acesso em: 03 mai. 2021. Citado 2 vezes nas páginas 22 e 23.
- APOLINÁRIO, J. P. A computação na era de sua criatividade: uma análise da produção científica do congresso internacional de criatividade computacional e um diálogo com a comunicação. 2018. Citado na página 16.
- ARRIGONI, M. de M. Debatendo os conceitos de caricatura, charge e cartum. 2011. Citado na página 21.
- BANSAL, H.; RATHORE, A. *Understanding and Implementing CycleGAN in TensorFlow*. 2017. Disponível em: <<https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/>>. Acesso em: 05 Out. 2021. Citado 2 vezes nas páginas 36 e 37.
- BARRETO, F. d. P. Artelligent: arte, inteligência artificial e criatividade computacional. 2016. Citado na página 14.
- BERGAMO, M. L.; SILVA, A. L. Doilies digitais: Um estudo da aplicação de criatividade computacional ao crochê. *DAT Journal*, v. 5, n. 1, p. 138–152, 2020. Citado 2 vezes nas páginas 14 e 15.
- BODEN, M. A. *et al.* *The creative mind: Myths and mechanisms*. [S.l.]: Psychology Press, 2004. Citado na página 14.
- BRITO, R. F. de; VANZIN, T.; ULBRICHT, V. R. Reflexões sobre o conceito de criatividade: sua relação com a biologia do conhecer. *Ciências & Cognição*, v. 14, n. 3, p. 204–213, 2009. Citado na página 10.
- BROWNLEE, J. *A Gentle Introduction to Generative Adversarial Networks (GANs)*. 2019. Disponível em: <<https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/g>>. Acesso em: 07 out. 2021. Citado 2 vezes nas páginas 35 e 38.
- CARAZO, C. *Caricaturas*. [S.l.]: Editorial Universidad de Costa Rica, 2003. Citado na página 18.
- CHEN, L.-C. *et al.* Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. Citado na página 41.

- COLTON, S.; WIGGINS, G. A. *et al.* Computational creativity: The final frontier? In: MONTPELIER. *Ecai*. [S.l.], 2012. v. 12, p. 21–26. Citado 3 vezes nas páginas 14, 15 e 51.
- CORRÊA, L. B. Carlos lacerda jornalista: repórter, gestor e teórico da imprensa. Universidade Federal do Rio de Janeiro, 2015. Citado na página 21.
- CREATIVITY, A. F. C. *Computational creativity*. 2017. Disponível em: <<https://computationalcreativity.net/iccc2017/>>. Acesso em: 07 mai. 2021. Citado na página 15.
- CRESWELL, A. *et al.* Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, IEEE, v. 35, n. 1, p. 53–65, 2018. Citado na página 34.
- DRESCH, A.; LACERDA, D. P.; JÚNIOR, J. A. V. A. *Design science research: método de pesquisa para avanço da ciência e tecnologia*. [S.l.]: Bookman Editora, 2015. Citado na página 12.
- FERNANDES, R. *Melhores aplicativos para transformar suas fotos em caricaturas*. 2019. Disponível em: <<https://www.techtudo.com.br/listas/2019/03/melhores-aplicativos-para-transformar-suas-fotos-em-caricaturas.ghtml>>. Acesso em: 30 mai. 2021. Citado na página 23.
- FILHO, L. d. F. C. L. *microcosmo da questão da arte na modernidade*. Tese (Doutorado) — PUC-Rio, 2006. Citado 2 vezes nas páginas 20 e 21.
- GOOCH, B.; REINHARD, E.; GOOCH, A. Human facial illustrations: Creation and psychophysical evaluation. *ACM Transactions on Graphics (TOG)*, ACM New York, NY, USA, v. 23, n. 1, p. 27–44, 2004. Citado na página 19.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep learning*. [S.l.]: MIT press, 2016. Citado na página 34.
- GOODFELLOW, I. J. *et al.* *Generative Adversarial Networks*. 2014. Citado na página 34.
- HAN, X. *et al.* Caricatureshop: Personalized and photorealistic caricature sketching. *IEEE transactions on visualization and computer graphics*, IEEE, v. 26, n. 7, p. 2349–2361, 2018. Citado 2 vezes nas páginas 31 e 32.
- HERNANDES, E. *et al.* Avaliação da ferramenta start utilizando o modelo tam e o paradigma gqm. In: *Proceedings of 7th Experimental Software Engineering Latin American Workshop (ESELAW 2010)*. [S.l.: s.n.], 2010. p. 30. Citado na página 29.
- HUGHES, A. *Learn to draw caricatures*. [S.l.]: HarperCollins Publishers, 1999. Citado na página 18.
- HUO, J. *et al.* Variation robust cross-modal metric learning for caricature recognition. In: *Proceedings of the on Thematic Workshops of ACM Multimedia 2017*. [S.l.: s.n.], 2017. p. 340–348. Citado na página 33.
- HUO, J. *et al.* Webcaricature: a benchmark for caricature recognition. In: *British Machine Vision Conference*. [S.l.: s.n.], 2018. Citado na página 33.

IBM. *The quest for AI creativity*. s/d. Disponível em: <<https://www.ihcriei.com.br/economiacriativa/pesquisa-pesquisaedesenvolvimento/tic/criatividade-computacional-um-passo-antes-da-inteligencia-artificial>>. Acesso em: 05 mai. 2021. Citado na página 16.

IH!CRIEI. *Criatividade computacional: a etapa antes da inteligência artificial*. 2021. Disponível em: <<https://www.ibm.com/watson/advantage-reports/future-of-artificial-intelligence/ai-creativity.htmlsection2>>. Acesso em: 05 mai. 2021. Citado 2 vezes nas páginas 15 e 17.

JR, J. F. N.; CHEN, M.; PURDIN, T. D. Systems development in information systems research. *Journal of management information systems*, Taylor & Francis, v. 7, n. 3, p. 89–106, 1990. Citado na página 12.

JÚNIOR, A. L. P. História entre o riso e a caricatura: uma proposta de interpretação teórico-metodológica das imagens satíricas com base nas obras de bruno paul, louis raemaekers e emmanuel poiré. *Diálogos-Revista do Departamento de História e do Programa de Pós-Graduação em História*, Universidade Estadual de Maringá, v. 20, n. 1, p. 73–85, 2016. Citado 2 vezes nas páginas 20 e 21.

KIRKPATRICK, D. *Toyota taps IBM Watson for personalized Rav4 campaign*. 2017. Disponível em: <<https://www.marketingdive.com/news/toyota-taps-ibm-watson-for-personalized-rav4-campaig>>. Acesso em: 05 mai. 2021. Citado na página 16.

KITCHENHAM, B. A.; CHARTERS, S. *Guidelines for performing Systematic Literature Reviews in Software Engineering*. [S.l.], 2007. Disponível em: <https://www.elsevier.com/_data/promis_misc/525444systematicreviewsguide.pdf>. Citado na página 28.

LACERDA, D. P. *et al.* Design science research: método de pesquisa para a engenharia de produção. *Gestão & produção*, SciELO Brasil, v. 20, p. 741–761, 2013. Citado na página 12.

LE, N. K. H.; WHY, Y. P.; ASHRAF, G. Shape stylized face caricatures. In: SPRINGER. *International Conference on Multimedia Modeling*. [S.l.], 2011. p. 536–547. Citado na página 19.

LEE, K.; BYATT, G.; RHODES, G. Caricature effects, distinctiveness, and identification: Testing the face-space framework. *Psychological science*, SAGE Publications Sage CA: Los Angeles, CA, v. 11, n. 5, p. 379–385, 2000. Citado na página 19.

LI, P. *et al.* 3d caricature generation by manifold learning. In: IEEE. *2008 IEEE International Conference on Multimedia and Expo*. [S.l.], 2008. p. 941–944. Citado 2 vezes nas páginas 30 e 31.

LI, W. *et al.* Carigan: Caricature generation through weakly paired adversarial learning. *Neural Networks*, Elsevier, v. 132, p. 66–74, 2020. Citado 3 vezes nas páginas 18, 31 e 32.

LIU, Z. *et al.* Deep learning face attributes in the wild. In: *Proceedings of International Conference on Computer Vision (ICCV)*. [S.l.: s.n.], 2015. Citado na página 39.

- MACHADO, M. *Dicas fundamentais para se criar uma boa caricatura*. 2013. Disponível em: <<https://www.desenhoonline.com/site/dicas-fundamentais-para-se-criar-uma-boa-caricatura/>>. Acesso em: 30 mai. 2021. Citado 2 vezes nas páginas 21 e 22.
- MARIANI. *Caricatura*. 2014. Disponível em: <<http://www.conteudoseducar.com.br/conteudos/arquivos/3274.pdf>>. Acesso em: 15 mai. 2021. Citado na página 23.
- MAURO, R.; KUBOVY, M. Caricature and face recognition. *Memory & Cognition*, Springer, v. 20, n. 4, p. 433–440, 1992. Citado na página 10.
- MENDES, A. I. F. *Caricatura e reconhecimento de faces*. Tese (Doutorado) — Universidade de São Paulo, 2007. Citado na página 19.
- NAIN, A. K. *CycleGAN*. 2020. Disponível em: <<https://github.com/keras-team/keras-io/blob/master/examples/generative/cyclegan.py>>. Acesso em: 12 Jul. 2021. Citado na página 40.
- PEREIRA, F. C. *Creativity and artificial intelligence: a conceptual blending approach*. [S.l.]: Walter de Gruyter, 2007. Citado na página 10.
- PIRES, C. O designer golem: um modelo computacional da criatividade em design. 2017. Citado 2 vezes nas páginas 14 e 50.
- RADFORD, A.; METZ, L.; CHINTALA, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. Citado 3 vezes nas páginas 35, 36 e 39.
- RHODES, G. *et al.* Coding spatial variations in faces and simple shapes: a test of two models. *Vision research*, Elsevier, v. 38, n. 15-16, p. 2307–2321, 1998. Citado 2 vezes nas páginas 18 e 19.
- SADIMON, S. B. *et al.* Computer generated caricature: A survey. In: IEEE. *2010 International Conference on Cyberworlds*. [S.l.], 2010. p. 383–390. Citado 2 vezes nas páginas 10 e 20.
- SIDDAVATAM, K. I.; SIDDAVATAM, I. A. Automatic caricature generation using text based input. In: SPRINGER. *International Conference on Parallel Distributed Computing Technologies and Applications*. [S.l.], 2011. p. 344–356. Citado 3 vezes nas páginas 18, 31 e 32.
- SIMON, H. A. *The sciences of the artificial*. [S.l.]: MIT press, 1996. Citado na página 12.
- SMITH, A.; MATEAS, M. Computational caricatures: Probing the game design process with ai. In: *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. [S.l.: s.n.], 2011. v. 7, n. 2. Citado 3 vezes nas páginas 18, 19 e 20.
- STRØMSTED, E. B. When caricature meets resistance. *Negotiating Journalism. Core values and cultural diversities*, p. 121–135, 2017. Citado 2 vezes nas páginas 18 e 20.

UJJWALKARN, K. *An Intuitive Explanation of Convolutional Neural Networks*. 2016. Disponível em: <<https://hardikbansal.github.io/CycleGANBlog/>>. Acesso em: 05 Out. 2021. Citado na página 36.

VELLAR, V. Criatividade computacional: Uma análise da utilização de técnicas computacionais criativas. *Dissertação de mestrado - Indústria Criativa*, Universidade FEEVALE, 2019. Citado 4 vezes nas páginas 10, 15, 16 e 17.

VERONESE, T. A importância da criatividade computacional para a literatura generativa: reflexões sobre arte, ciência e tecnologia na ciberliteratura. *Revista Scientiarum Historia*, v. 1, p. 8–8, 2017. Citado na página 10.

WEI, L. *et al.* Application of facial feature localization using constrained local models in template-based caricature synthesis. In: IEEE. *2009 2nd International Congress on Image and Signal Processing*. [S.l.], 2009. p. 1–5. Citado 3 vezes nas páginas 30, 31 e 32.

WIKIART, V. A. E. *Sheet of caricatures*. 2011. Disponível em: <<https://www.wikiart.org/en/annibale-carracci/sheet-of-caricatures>>. Acesso em: 30 mai. 2021. Citado na página 19.

ZHU, J.-Y. *et al.* Unpaired image-to-image translation using cycle-consistent adversarial networks. In: *Proceedings of the IEEE international conference on computer vision*. [S.l.: s.n.], 2017. p. 2223–2232. Citado 2 vezes nas páginas 37 e 38.

Apêndices

APÊNDICE A – FORMULÁRIO DE SOLICITAÇÃO DO DATASET

WebCaricature Dataset Release Agreement

Introduction

The WebCaricature Dataset can only be used for scientific research purposes. It was collected by the R&L Group of Nanjing University, China. Any individuals or groups may use this Dataset free of charge for the purposes stated above.

Rules

1. **Redistribution:** Without prior permission from the R&L Group, the Dataset, either entirely or partly, must not be further distributed, published, copied, or disseminated in any way or form, no matter for profitable use or not, including further distribution to a different department or organization in the same system.
2. **Modification:** Without prior permission from the R&L Group, the Dataset, either entirely or partly, is not allowed to be modified.
3. **Commercial use:** Without prior permission from the R&L Group, the Dataset, either entirely, partly, or in any format derived thereof, is not allowed for commercial use.
4. **Publication requirements:** In no case should the images be used in any way that may possibly cause dispute, embarrassment or mental anguish to any of the subjects in the Dataset.
5. **Acknowledgement:** The following references should be cited in all documents and papers that report experimental results conducted based on this Dataset:
[1] Jing Huo, Wenbin Li, Yinghuan Shi, Yang Gao and Hujun Yin, "WebCaricature: A Benchmark for Caricature Recognition", *British Machine Vision Conference*, 2018.
[2] Jing Huo, Yang Gao, Yinghuan Shi, Hujun Yin, "Variation Robust Cross-Modal Metric Learning for Caricature Recognition", *ACM Multimedia Thematic Workshops*, 2017: 340-348.
6. **Indemnification:** All persons using the Dataset agree to indemnify, defend and cause no harm to the R&L Group and its employees and agents, individually and collectively, from and against any loss, expense and damage. Users shall pay for all damages, claims, judgments or expenses caused by misuse of this Dataset.

Note: Signatory must have the legal authority on behalf of the licensee to sign the licensing agreements with the R&L Group. Licensee's legal staff should review the terms and conditions of this agreement and execute it. Agreements signed by individuals without this authority are not valid and will be discarded.

Name and title: Suelem Kleinkauf

Job title: PHOTO-BASED CARICATURE GENERATION PROTOTYPE

Nationality: Brazil

Organization: University Feevale

Department/Laboratory: Information Systems

Address: RS-239, 2755 - Vila Nova, Novo Hamburgo - RS, 93525-075

Email: suelemkleinkauf@gmail.com

Telephone: 55 51 999265155

Signature and date: Suelem Kleinkauf 09/10/2021

Please scan the completed agreement and send it to huojing1989@gmail.com or huojing@nju.edu.cn. A notification email will be sent to the email address and downloading of the Dataset will be authorized once the procedure is approved.

APÊNDICE B – E-MAIL DE AUTORIZAÇÃO PARA USO DO DATASET

07/11/2021 15:09

Gmail - Request Access to WebCaricature



Suelem Kleinkauf <suelemkleinkauf@gmail.com>

Request Access to WebCaricature

Jing Huo <huojing1989@gmail.com>
Para: Suelem Kleinkauf <suelemkleinkauf@gmail.com>

13 de setembro de 2021 09:22

Hi,

Thank you for your interest in our work. Your application to use the dataset is approved.

You can download the dataset via this link: [REDACTED] [Link de acesso omitido](#)

Should you have any problems about downloading or using the dataset, please do not hesitate to contact us.
Suggestions are also welcome.

Best,
Jing

Suelem Kleinkauf <suelemkleinkauf@gmail.com> 于2021年9月11日周六 上午7:05写道:
[Texto das mensagens anteriores oculto]

—
Jing Huo
Department of Computer Science and Technology
State Key Laboratory for Novel Software Technology
Nanjing University
Email: huojing1989@gmail.com, huojing@nju.edu.cn

APÊNDICE C – TABELA DE *LOSS* TREINAMENTO COM 100 ÉPOCAS

Época	Loss Gerador G	Loss Gerador F	Loss Discriminador X	Loss Discriminador Y
6	2,5318	2,5318	2.026	1.994
7	2,4558	2,4558	2.048	1.957
8	2,4573	2,4573	2.067	1.930
9	2,4265	2,4265	2.113	1.901
10	2,4081	2,4081	2.088	1.885
11	2,4084	2,4084	2.072	1.878
12	2,4184	2,4184	2.038	1.847
13	2,4302	2,4302	1.996	1.853
14	2,4078	2,4078	2.004	1.867
15	2,3635	2,3635	1.988	2.023
16	2,3345	2,3345	1.989	2.002
17	2,3251	2,3251	1.947	2.001
18	2,2975	2,2975	1.961	2.010
19	2,2909	2,2909	1.956	2.018
20	2,2624	2,2624	1.943	2.034
21	2,2396	2,2396	1.932	2.011
22	2,2335	2,2335	1.952	2.008
23	2,2505	2,2505	1.952	1.983
24	2,2323	2,2323	1.939	1.983
25	2,2442	2,2442	1.954	1.959
26	2,2134	2,2134	1.907	1.957
27	2,2209	2,2209	1.934	1.941
28	2,2239	2,2239	1.928	1.942
29	2,2095	2,2095	1.900	1.918
30	2,2209	2,2209	1.897	1.904
31	2,2307	2,2307	1.889	1.868
32	2,2265	2,2265	1.891	1.871
33	2,2001	2,2001	1.869	1.848
34	2,2033	2,2033	1.886	1.828
35	2,2068	2,2068	1.866	1.828
36	2,236	2,236	1.878	1.799
37	2,2213	2,2213	1.825	1.783
38	2,1988	2,1988	1.839	1.784
39	2,2484	2,2484	1.819	1.744
40	2,218	2,218	1.804	1.744
41	2,2249	2,2249	1.826	1.715
42	2,2621	2,2621	1.781	1.704
43	2,2165	2,2165	1.784	1.700
44	2,2469	2,2469	1.760	1.658
45	2,2578	2,2578	1.745	1.659
46	2,2439	2,2439	1.750	1.647
47	2,2233	2,2233	1.741	1.620
48	2,2416	2,2416	1.747	1.595
49	2,2749	2,2749	1.714	1.590
50	2,2325	2,2325	1.757	1.600
51	2,2534	2,2534	1.675	1.590
52	2,2625	2,2625	1.715	1.557

53	2,2505	2,2505	1.711	1.547
54	2,26	2,26	1.714	1.526
55	2,2655	2,2655	1.731	1.513
56	2,2755	2,2755	1.711	1.513
57	2,3055	2,3055	1.689	1.476
58	2,2914	2,2914	1.726	1.488
59	2,2939	2,2939	1.718	1.470
60	2,3035	2,3035	1.641	1.444
61	2,2994	2,2994	1.686	1.456
62	2,3219	2,3219	1.697	1.432
63	2,3287	2,3287	1.664	1.421
64	2,3217	2,3217	1.674	1.406
65	2,3449	2,3449	1.658	1.394
66	2,3476	2,3476	1.674	1.367
67	2,3243	2,3243	1.649	1.388
68	2,3587	2,3587	1.653	1.385
69	2,3349	2,3349	1.643	1.363
70	2,3579	2,3579	1.644	1.319
71	2,3641	2,3641	1.609	1.342
72	2,3662	2,3662	1.628	1.319
73	2,3722	2,3722	1.618	1.342
74	2,3796	2,3796	1.569	1.331
75	2,3819	2,3819	1.590	1.330
76	2,3925	2,3925	1.569	1.317
77	2,3956	2,3956	1.563	1.300
78	2,3946	2,3946	1.562	1.292
79	2,4075	2,4075	1.555	1.292
80	2,4078	2,4078	1.564	1.293
81	2,3854	2,3854	1.555	1.282
82	2,4177	2,4177	1.546	1.277
83	2,4194	2,4194	1.549	1.268
84	2,4012	2,4012	1.504	1.289
85	2,4161	2,4161	1.495	1.257
86	2,4163	2,4163	1.505	1.268
87	2,4104	2,4104	1.495	1.259
88	2,4151	2,4151	1.491	1.254
89	2,4137	2,4137	1.476	1.267
90	2,4261	2,4261	1.482	1.267
91	2,4084	2,4084	1.477	1.261
92	2,4153	2,4153	1.461	1.248
93	2,4326	2,4326	1.468	1.243
94	2,4289	2,4289	1.459	1.255
95	2,4246	2,4246	1.440	1.251
96	2,4355	2,4355	1.448	1.238
97	2,4281	2,4281	1.441	1.251
98	2,4375	2,4375	1.447	1.223
99	2,4289	2,4289	1.435	1.256
100	2,4245	2,4245	1.432	1.260

**APÊNDICE D – TABELA DE *LOSS* TREINAMENTO COM 100
ÉPOCAS E *BATCH_SIZE* 4**

Época	Loss Gerador G	Loss Discriminador X	Loss Gerador F	Loss Discriminador Y
2	3,5463	1.871	3.1984	1.490
3	3,3409	1.634	2.8446	1.408
4	3,1172	1.795	2.6650	1.541
5	2,9223	1.880	2.5413	1.796
6	2,7757	1.936	2.3976	1.761
7	2,6816	1.971	2.3344	1.880
8	2,5908	1.991	2.2351	1.889
9	2,5455	2.012	2.1612	1.915
10	2,4644	2.011	2.1254	2.106
11	2,4644	2.029	2.0933	1.944
12	2,4646	2.022	2.1140	1.956
13	2,5116	2.050	2.0807	1.933
14	2,45	2.053	2.0751	1.950
15	2,4398	2.012	2.0355	2.738
16	2,1721	2.392	1.9219	2.382
17	2,1673	2.168	1.8792	2.321
18	2,2225	2.083	1.9442	2.318
19	2,2238	2.045	1.9668	2.300
20	2,2653	2.069	1.9805	2.243
21	2,2866	2.074	1.9348	2.177
22	2,3126	2.077	1.9500	2.112
23	2,2808	2.043	1.9736	2.033
24	2,3146	2.057	1.9646	2.044
25	2,3074	2.015	1.9813	1.980
26	2,2902	1.996	1.9564	1.958
27	2,3244	1.977	1.9941	1.934
28	2,3106	1.943	1.9729	1.954
29	2,2945	1.955	1.9686	1.944
30	2,3044	1.920	2.1978	1.861
31	2,2866	1.925	2.2018	1.904
32	2,2919	1.923	1.9826	1.900
33	2,3076	1.912	1.9877	1.916
34	2,2737	1.914	1.9657	1.991
35	2,2325	1.859	2.0164	1.990
36	2,2216	1.813	2.0753	2.088
37	2,209	1.847	2.0630	2.063
38	2,1912	1.838	2.0447	2.041
39	2,1986	2.736	2.1408	2.040
40	2,0712	2.483	1.9210	2.024
41	2,0762	2.452	1.8335	2.001
42	2,0612	2.421	1.8717	2.016
43	2,0649	2.409	1.8373	2.008
44	2,0733	2.376	1.8383	1.997
45	2,1007	2.352	1.8420	1.984
46	2,0839	2.272	1.8780	1.976
47	2,1587	2.096	1.9127	1.927
48	2,1475	1.928	1.9210	1.917

49	2,118	1.829	1.9753	1.919
50	2,1206	1.813	1.9987	1.922
51	2,1664	1.786	1.9788	1.899
52	2,1649	1.776	1.9842	1.882
53	2,1893	1.779	1.9847	1.836
54	2,1628	1.785	1.9958	1.868
55	2,1452	1.760	1.9945	1.846
56	2,1852	1.776	1.9665	1.812
57	2,1806	1.748	2.0215	1.839
58	2,1542	1.741	2.2011	1.829
59	2,122	1.740	1.9957	1.838
60	2,1566	1.760	1.9599	1.774
61	2,1404	1.745	2.2001	1.808
62	2,1573	1.753	1.9813	1.800
63	2,2071	1.757	1.9734	1.748
64	2,1512	1.716	1.9873	1.771
65	2,171	1.738	1.9717	1.748
66	2,1561	1.750	1.9586	1.747
67	2,149	1.729	1.9705	1.737
68	2,1455	1.702	1.9945	1.762
69	2,1566	1.706	1.9984	1.739
70	2,1608	1.705	2.1986	1.710
71	2,1736	1.693	2.1997	1.714
72	2,1804	1.688	2.0160	1.706
73	2,1622	1.705	1.9912	1.697
74	2,1777	1.690	1.9938	1.683
75	2,1544	1.687	1.9831	1.671
76	2,1748	4.638	2.4185	1.661
77	2,0311	2.586	1.8690	1.676
78	2,4365	2.532	1.7646	4.717
79	1,7506	2.521	1.5226	2.572
80	1,6976	2.517	1.5498	2.535
81	1,684	2.510	1.5465	2.516
82	1,699	2.497	1.5736	2.506
83	1,701	2.498	1.5376	2.485
84	1,6959	2.447	1.6251	2.494
85	1,737	2.215	1.6385	2.461
86	1,8105	1.785	1.7839	2.321
87	1,9694	1.690	1.8610	1.856
88	2,0159	1.633	1.8946	1.668
89	2,0956	1.665	1.8874	1.597
90	2,0788	1.681	1.9118	1.625
91	2,0962	1.652	1.9241	1.625
92	2,1018	1.672	1.9220	1.588
93	2,098	1.681	1.9207	1.633
94	2,089	1.674	1.9522	1.630
95	2,1147	1.665	1.9723	1.615
96	2,0872	1.648	1.9718	1.646

97	2,1496	1.648	1.9326	1.581
98	2,1184	1.646	1.9541	1.580
99	2,1174	1.653	1.9623	1.603
100	2,1349	1.622	1.9929	1.580

APÊNDICE E – TABELA DE *LOSS* TREINAMENTO COM 150 ÉPOCAS

Época	Loss Gerador G	Loss Gerador F	Loss Discriminador X	Loss Discriminador Y
55	2.3761	2.1639	1.524	1.533
56	2.3546	2.1675	1.502	1.529
57	2.3767	2.1731	1.518	1.500
58	2.3700	2.1944	1.480	1.481
59	2.3802	2.1852	1.473	1.497
60	2.3380	2.1975	1.471	1.510
61	2.3825	2.2196	1.451	1.477
62	2.3821	2.2185	1.454	1.469
63	2.3807	2.2107	1.408	1.462
64	2.3776	2.2140	1.428	1.419
65	2.3870	2.2067	1.396	1.430
66	2.3846	2.2093	1.423	1.421
67	2.3952	2.2071	1.388	1.400
68	2.3620	2.2176	1.435	1.436
69	2.3996	2.2283	1.372	1.413
70	2.3776	2.2318	1.357	1.403
71	2.3803	2.2420	1.331	1.440
72	2.3900	2.2221	1.373	1.372
73	2.3700	2.2309	1.369	1.401
74	2.3684	2.2250	1.364	1.410
75	2.3813	2.2296	1.354	1.360
76	2.3751	2.2501	1.287	1.387
77	2.3642	2.2542	1.331	1.392
78	2.3607	2.2609	1.313	1.385
79	2.3593	2.2299	1.319	1.365
80	2.3703	2.2142	1.359	1.362
81	2.3643	2.2365	1.308	1.353
82	2.3765	2.2153	1.352	1.347
83	2.3691	2.2127	1.319	1.333
84	2.3649	2.2254	1.323	1.371
85	2.3642	2.2163	1.331	1.320
86	2.3458	2.1930	1.361	1.368
87	2.3704	2.2411	1.254	1.339
88	2.3627	2.1954	1.305	1.322
89	2.3680	2.1918	1.314	1.353
90	2.3615	2.1833	1.333	1.345
91	2.3738	2.1804	1.312	1.315
92	2.3498	2.1952	1.296	1.351
93	2.3636	2.1912	1.299	1.355
94	2.3505	2.1706	1.308	1.337
95	2.3560	2.2006	1.285	1.328
96	2.3463	2.1808	1.306	1.336
97	2.3555	2.1883	1.288	1.329
98	2.3792	2.1783	1.320	1.323
99	2.3477	2.1678	1.277	1.351
100	2.3538	2.1732	1.292	1.322
101	2.3425	2.1661	1.297	1.355
102	2.3476	2.2144	1.187	1.319
103	2.3448	2.1566	1.304	1.350

104	2.3533	2.1524	1.323	1.317
105	2.3285	2.1466	1.290	1.355
106	2.3452	2.1541	1.271	1.356
107	2.3431	2.1327	1.298	1.336
108	2.3174	2.1584	1.261	1.398
109	2.3281	2.1624	1.286	1.350
110	2.3380	2.1461	1.286	1.317
111	2.3248	2.1459	1.250	1.355
112	2.3308	2.1385	1.286	1.344
113	2.3358	2.1248	1.288	1.356
114	2.3157	2.1395	1.296	1.369
115	2.3188	2.1353	1.284	1.366
116	2.3113	2.1213	1.282	1.353
117	2.3273	2.1307	1.274	1.373
118	2.3239	2.1226	1.281	1.340
119	2.3279	2.1272	1.283	1.372
120	2.3266	2.0972	1.309	1.340
121	2.3037	2.1124	1.273	1.378
122	2.3214	2.1072	1.287	1.357
123	2.3228	2.0978	1.298	1.334
124	2.3177	2.1081	1.284	1.372
125	2.3291	2.1035	1.258	1.319
126	2.3057	2.1044	1.277	1.385
127	2.3223	2.1185	1.255	1.353
128	2.3156	2.1085	1.257	1.351
129	2.3008	2.1178	1.245	1.352
130	2.3080	2.1130	1.250	1.349
131	2.3031	2.1171	1.246	1.359
132	2.3066	2.1240	1.218	1.335
133	2.3011	2.1282	1.232	1.358
134	2.3179	2.0870	1.286	1.336
135	2.2952	2.0970	1.240	1.373
136	2.2955	2.0836	1.279	1.359
137	2.2937	2.0795	1.279	1.338
138	2.2944	2.0971	1.236	1.347
139	2.3189	2.0863	1.253	1.314
140	2.2794	2.0860	1.254	1.378
141	2.2935	2.0580	1.305	1.345
142	2.2844	2.0841	1.274	1.387
143	2.3083	2.0796	1.236	1.305
144	2.2780	2.0597	1.298	1.373
145	2.3007	2.0642	1.277	1.329
146	2.2885	2.0881	1.236	1.361
147	2.2794	2.0769	1.279	1.377
148	2.3001	2.0562	1.276	1.320
149	2.2893	2.0694	1.263	1.359
150	2.2834	2.0642	1.271	1.354

APÊNDICE F – TABELA DE *LOSS* TREINAMENTO COM IMAGENS NEUTRALIZADAS

Época	Loss Gerador G	Loss Gerador F	Loss Discriminador X	Loss Discriminador Y
39	1.5460	1.5778	1.854	1.826
40	1.6237	1.6478	1.823	1.770
41	1.5452	1.5954	1.846	1.807
42	1.6241	1.7655	1.788	1.801
43	1.5827	1.6004	1.822	1.750
44	1.5860	1.6262	1.830	1.718
45	1.5600	1.6414	1.824	1.773
46	1.5715	1.6425	1.826	1.737
47	1.6281	1.6247	1.835	1.679
48	1.5930	1.6412	1.820	1.679
49	1.6085	1.5915	1.826	1.637
50	1.6230	1.6110	1.817	1.636
51	1.6186	1.6363	1.832	1.634
52	1.6287	1.6475	1.833	1.639
53	1.6089	1.6295	1.808	1.625
54	1.7251	1.6104	1.800	1.549
55	1.6322	1.6523	1.816	1.600
56	1.6528	1.6484	1.812	1.555
57	1.7281	1.6593	1.807	1.523
58	1.8759	1.7227	1.746	1.416
59	1.7378	1.6804	1.802	1.501
60	1.7324	1.6411	1.766	1.436
61	1.7570	1.6964	1.821	1.495
62	1.7084	1.6524	1.784	1.476
63	1.7395	1.7193	1.773	1.447
64	1.7613	1.6570	1.836	1.392
65	1.7148	1.6530	1.820	1.458
66	1.7439	1.6901	1.834	1.394
67	1.7565	1.6400	1.794	1.371
68	1.7783	1.6458	1.785	1.323
69	1.8180	1.6758	1.742	1.341
70	1.9315	1.7419	1.785	1.301
71	1.8611	1.8029	1.762	1.303
72	1.9463	1.6095	1.785	1.177
73	2.0118	1.7287	1.772	1.236
74	1.8466	1.6467	1.709	1.215
75	1.8368	1.6036	1.799	1.273
76	1.8326	1.5871	1.786	1.225
77	1.8900	1.6017	1.842	1.245
78	1.8454	1.6079	1.833	1.253
79	1.9420	1.8685	1.734	1.244
80	2.1112	1.7019	1.779	1.145
81	1.9162	1.6521	1.763	1.164
82	2.0190	1.6844	1.805	1.130
83	1.9338	1.6151	1.801	1.110
84	1.9248	1.5869	1.805	1.110
85	1.9010	1.6035	1.786	1.134
86	1.8969	1.5883	1.778	1.106
87	2.1484	1.7325	1.730	1.068

88	1.9498	1.5554	1.791	1.043
89	2.0837	1.7005	1.764	1.092
90	1.9972	1.7043	1.743	1.090
91	2.2000	1.6800	1.780	991
92	2.0883	1.6424	1.806	1.020
93	1.9598	1.6476	1.771	1.049
94	2.0476	1.7149	1.764	1.025
95	2.1988	1.6508	1.806	1.027
96	2.0362	1.6400	1.809	970
97	2.0312	1.6215	1.772	986
98	2.0465	1.6423	1.786	965
99	1.9945	1.6591	1.785	972
100	2.2608	1.6114	1.778	802