

FEDERAL UNIVERSITY OF TECHNOLOGY - PARANÁ

RODRIGO TCHALSKI DA SILVA

**COMPUTER VISION METHODS FOR TATTOO DETECTION,
LOCATION AND CLASSIFICATION**

DISSERTATION

CURITIBA

2022

RODRIGO TCHALSKI DA SILVA

**COMPUTER VISION METHODS FOR TATTOO DETECTION,
LOCATION AND CLASSIFICATION**

**Métodos de Visão Computacional para Detecção, Localização e
Classificação de Tatuagens**

Dissertation presented to the Graduate Program in Electrical and Computer Engineering of the Federal University of Technology - Paraná as part of the fulfillment of the requirements for the title of Master in Electrical Engineering and Industrial Informatics.

Advisor: Prof. Dr. Heitor Silvério Lopes

CURITIBA

2022



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es).

Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.



**Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Curitiba**



RODRIGO TCHALSKI DA SILVA

COMPUTER VISION METHODS FOR TATTOO DETECTION, LOCATION AND CLASSIFICATION

Trabalho de pesquisa de mestrado apresentado como requisito para obtenção do título de Mestre Em Ciências da Universidade Tecnológica Federal do Paraná (UTFPR). Área de concentração: Engenharia De Computação.

Data de aprovação: 31 de Maio de 2022

Dr. Heitor Silverio Lopes, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Andre Eugenio Lazzaretti, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Nelson Marcelo Romero Aquino, Doutorado - Universite de Lorraine

Dr. Rodrigo Minetto, Doutorado - Universidade Tecnológica Federal do Paraná

Documento gerado pelo Sistema Acadêmico da UTFPR a partir dos dados da Ata de Defesa em 04/07/2022.

To my eternal companion Emilyn and my
children Isabella, Gabriela and Henrique, for the
moments of absence.

ACKNOWLEDGEMENTS

My thanks to the professors at UTFPR for their excellence and dedication to teaching science in our country, and for all their contribution to my learning during this period.

My special thanks to my supervisor, Prof. Dr. Heitor Silvério Lopes for his support, patience and dedication in my guidance. His guidance, advice and directions were fundamental for achieving this result.

My deepest thanks to my wife Emilyn and my children Isabella, Gabriela and Henrique for their patience and support during this period of study. Your love has always been fundamental to everything I do.

To the colleagues in the department, who helped directly and indirectly in the completion of this work.

"BE PROACTIVE. Being proactive is more than taking the initiative. It is recognizing that we are responsible for our own choices and that we have the freedom to choose based on principles and values rather than circumstances and conditions. Proactive people are agents of change and choose not to be victims, not to be reactive, or to blame others." (COVEY, 2004)

RESUMO

SILVA, Rodrigo Tchalski da. **Métodos de Visão Computacional para Detecção, Localização e Classificação de Tatuagens**. 2022. 95 f. Dissertation (Master's Degree em Electrical Engineering and Industrial Informatics) – Federal University of Technology - Paraná. Curitiba, 2022.

As tatuagens ainda são pouco exploradas como fator biométrico para identificação humana, principalmente na segurança pública, onde elas podem desempenhar um papel importante na identificação de criminosos, vítimas ou outras pessoas de interesse. As tatuagens são classificadas como biometria suave, pois não são permanentes e podem mudar ao longo do tempo, diferentemente dos traços biométricos rígidos (impressão digital, íris, DNA, etc.). Desta forma, o objetivo principal deste trabalho é aplicar métodos de visão computacional e transferência de aprendizado para os problemas de detecção, localização e classificação de tatuagens em imagens. Dada a escassez de bases de dados disponíveis na literatura para estes problemas, foram criadas bases de dados anotadas específicas para cada um dos problemas aqui abordados. Para o problema de detecção de tatuagens foi apresentado um modelo de aprendizado profundo baseado em transferência de aprendizado. Também foi aplicada a técnica de data augmentation para melhorar a diversidade dos conjuntos de treinamento para obter uma melhor precisão de classificação, e experimentos comparativos foram feitos para avaliar a diversidade de imagens nos conjuntos de dados e a precisão do modelo proposto. Para o problema de localização de tatuagens foi apresentada uma abordagem retreinando a rede Mask R-CNN com uma base de dados de tatuagens, e um fine tuning foi realizado na rede com o objetivo de encontrar o conjunto de parâmetros que apresentasse melhores resultados no treinamento da rede. Para o problema de classificação de tatuagens o modelo proposto foi também baseado na utilização de redes profundas com transferência de aprendizado para classificar um conjunto de 40 categorias de tatuagens, muitas delas com significado prático para segurança pública. A técnica de data augmentation também foi utilizada para melhorar a diversidade e robustez dos dados de treinamento. Na detecção de tatuagens os resultados foram muito promissores, alcançando uma precisão de 95,1% no conjunto de teste e um F1-score de 0,79 em um conjunto de dados externo que, no geral, foram satisfatórios, dada a complexidade do problema. Na localização de tatuagens os resultados alcançaram uma precisão média de 89,3%, mostrando que a rede Mask R-CNN possui grande capacidade de adaptação para o ambiente de tatuagens, além de ser realizada uma análise qualitativa que ajudou a entender como as características das imagens e das anotações tem influência sobre os resultados. Na classificação de tatuagens, os resultados alcançaram 85,48% de acurácia ao utilizar validação cruzada e data augmentation, mostrando que a abordagem de transferência de aprendizado adotada tem boa capacidade para este problema. Trabalhos futuros incluirão melhorar a qualidade e o volume das bases de dados, realizar um estudo mais profundo sobre o ajuste fino de parâmetros das redes, e estudos de técnicas de mundo aberto para classificação de tatuagens, além de desenvolvimento de modelos para outros problemas que compõem o sistema de reconhecimento de tatuagens.

Palavras-chave: Detecção de tatuagem. Localização de tatuagem. Classificação e tatuagem. Visão computacional. Transferência de aprendizado.

ABSTRACT

SILVA, Rodrigo Tchalski da. **Computer Vision Methods for Tattoo Detection, Location and Classification**. 2022. 95 p. Dissertation (Master's Degree in Electrical Engineering and Industrial Informatics) – Federal University of Technology - Paraná. Curitiba, 2022.

Tattoos are still poorly explored as a biometric factor for human identification, especially in law enforcement, where they can play an important role in identifying criminals, victims or other persons of interest. Tattoos are classified as soft biometrics as they are not permanent and can change over time, unlike hard biometric traits (fingerprint, iris, DNA, etc.). In this way, the main objective of this work is to apply computer vision methods and transfer learning to the problems of tattoo detection, location and classification in images. Given the scarcity of datasets available in the literature for these problems, specific annotated datasets were created for each problem addressed here. For the tattoo detection problem, a deep learning model based on transfer learning was presented. Data augmentation technique was also applied to improve the diversity of the training sets to obtain a better classification accuracy, and comparative experiments were carried out to evaluate the diversity of images in the data sets and the accuracy of the proposed model. For the tattoo location problem, an approach was presented by retraining the Mask R-CNN network with a tattoo dataset, and a fine-tuning was performed on the network to find the set of parameters that presented the best results in training the network. For the tattoo classification problem, the proposed model was also based on using deep networks with transfer learning to classify a set of 40 tattoo categories, many of them with practical meaning for law enforcement. Data augmentation technique was also used to improve the diversity and robustness of the training data. In tattoo detection, the results were very promising, achieving an accuracy of 95.1% in the test dataset and an F1-score of 0.79 in an external dataset, which, in general, were satisfactory, given the complexity of the problem. In tattoos location, the results reached an average accuracy of 89.3%, showing that the Mask R-CNN network has great adaptability to the tattoo environment, in addition to performing a qualitative analysis that helped to understand how the characteristics of images and annotations influence the results. In tattoos classification, the results reached accuracy of 85.24% when using cross validation and data augmentation, showing that the transfer learning approach adopted has good capacity for this problem. Future work will include improving the quality and volume of the databases, conducting a more in-depth study on the fine-tuning of network parameters, and studies of open-world techniques for classifying tattoos, as well as developing models for other problems that compose the tattoo recognition roadmap.

Keywords: Tattoo detection. Tattoo location. Tattoo classification. Computer vision. Transfer learning.

LIST OF FIGURES

Figure 1 – Computer Vision in Artificial Intelligence field.	18
Figure 2 – Approaches to computer vision methods.	19
Figure 3 – A rough timeline of some of the most active topics of research in computer vision.	20
Figure 4 – Simple Neural Network and Deep Neural Network (Deep Learning).	21
Figure 5 – Inside a convolutional network.	22
Figure 6 – Example architecture of a Convolutional Neural Networks (CNN) for a computer vision task (object detection).	22
Figure 7 – Sketch map of instances-based deep transfer learning.	24
Figure 8 – Sketch map of mapping-based deep transfer learning.	24
Figure 9 – Sketch map of network-based deep transfer learning.	24
Figure 10 – Sketch map of adversarial-based deep transfer learning.	25
Figure 11 – A taxonomy of image data augmentation.	26
Figure 12 – Data augmentation samples.	27
Figure 13 – Data augmentation samples with Cycle Generative Adversarial Netss (GANs).	27
Figure 14 – Region-based Convolutional Neural Networks (R-CNN) architecture.	28
Figure 15 – Fast R-CNN architecture.	28
Figure 16 – Faster R-CNN architecture. Left: Faster R-CNN is a single, unified network for object detection. Center: RPN. Right: Example detections using Region Proposal Network (RPN) proposals on PASCAL VOC 2007 dataset test.	29
Figure 17 – Mask R-CNN framework.	29
Figure 18 – The You Only Look Once (YOLO) Detection System.	30
Figure 19 – Tattoo recognition roadmap.	31
Figure 20 – Number of relevant publications by category and year.	33
Figure 21 – Example of images, used for the detection task, of people with and without tattoos. Image "a" is an example of a well-behaved situation, where the tattoo is a well-defined, and the surrounding background is a clear skin. The Image "b" there are multiple tattoos, and a bounding-box of them will include other areas of the body where there is no tattoo. Also, the letters of the hat may disturb the recognition process. In image "c" the painting in the backward has similar patterns to the tattoo, and they are close to each other. The image "d" is a good example of non-tattoo, easily identified. In the image "e", the design of the clothes in contrast with the skin may be confused as a tattoo. In image "f" painted nails may be confused as a tattoo.	34
Figure 22 – Examples of tattoo location.	37
Figure 23 – Example of tattoos to be classified. Items "a", "b" and "c" shows images with one label, like a butterfly, a cat and a flower. Images "d", "e" and "f" shows tattoos with multiple labels, like a gun and flowers; a microphone, a person, a car and sunglasses; a skull, flowers, a heart, a diamond and a cross.	39
Figure 24 – Proposed tattoo detection transfer-learning model.	41
Figure 25 – Samples of tattoo and non tattoo images in train datasets.	43
Figure 26 – Mask R-CNN for Tattoo Location Architecture.	44
Figure 27 – Computing the Intersection over Union (IoU) is as simple as dividing the area of overlap between the bounding boxes by the area of union.	45

Figure 28 – Tattoo annotation system.	46
Figure 29 – Sample of different tattoo styles and proportion. Image “a” represents a color tattoo, image “b” represents an outline tattoo, image “c” represents a monochromatic tattoo, image “d” represents a large size tattoo, image “e” represents a medium size tattoo, image “f” represents a small size tattoo and image “g” represents a textual tattoo.	47
Figure 30 – Proposed tattoo classification transfer-learning model.	48
Figure 31 – Sample of tattoo classes.	49
Figure 32 – Confusion matrices for the experiments with an external dataset.	54
Figure 33 – Confusion matrices for different training and testing datasets.	56
Figure 34 – Examples of non-tattoos wrongly classified as tattoos.	56
Figure 35 – Examples of tattoos wrongly classified as non-tattoos.	57
Figure 36 – Loss to tattoo dataset with default Mask R-CNN parameters.	58
Figure 37 – Loss to tattoo dataset with fine tuning Mask R-CNN.	59
Figure 38 – Sample of good location bounding boxes.	61
Figure 39 – Sample of bad location bounding boxes.	61
Figure 40 – Accuracy and loss curves to tattoo classification initial model.	63
Figure 41 – Final tattoo classification dense network.	63
Figure 42 – Accuracy and loss curves to tattoo classification model with cross validation.	64
Figure 43 – Accuracy and loss curves to tattoo classification model with cross validation and data augmentation.	65
Figure 44 – Sample tattoo images of wizard, lion and tiger.	67
Figure 45 – Example of tattoo classes not trained in the proposed model.	68
Figure 46 – Example of tattoo with more than one class or tattoo.	69
Figure 47 – Images without tattoos.	71
Figure 48 – Retrieval results for tattoo classification initial model.	92
Figure 49 – Retrieval results for tattoo classification model with cross validation.	93
Figure 50 – Retrieval results for final tattoo classification model with cross validation and data augmentation.	94

LIST OF TABLES

Table 1 – Comparison between some common biometrics features and tattoos (✓ means “Yes”, ✗ means “No”, and “–” means “Does not apply”).	15
Table 2 – Tattoo Detection Published Results.	34
Table 3 – Tattoo datasets referenced in the tattoo detection bibliography.	35
Table 4 – Papers approaching the tattoo location problem.	37
Table 5 – Tattoo Classification Published Results.	39
Table 6 – Dataset for Tattoo Location - TattLocA.	47
Table 7 – Dataset for Tattoo Location - TattLocB.	47
Table 8 – Results for feature extractors evaluation.	52
Table 9 – Results for the data augmentation tests.	53
Table 10 – Results for the classification using an external dataset.	54
Table 11 – Hyper-parameters tested for Mask R-CNN tattoo location.	59
Table 12 – Final results for Mask R-CNN tattoo location.	60
Table 13 – Classes with the worst classification accuracy.	66
Table 14 – Sample of results of unknown classes classification.	68
Table 15 – Results for classification in multi classes tattoos.	70
Table 16 – Hyper-parameters tested in Mask R-CNN tattoo location.	86
Table 17 – Mask R-CNN Default Parameters.	91
Table 18 – Results of the classification of unknown classes.	95

LIST OF ACRONYMS

INITIALISM

AI	Artificial Intelligence
CNN	Convolutional Neural Networks
CV	Computer vision
DNN	Deep Neural Network
dvAP	standard deviation Average Precision
GAN	Generative Adversarial Nets
GMM	Gaussian Mixture Models
HOG	Histogram of Oriented Gradients
IoU	Intersection over Union
kNN	k-Nearest Neighbors
mAP	mean Average Precision
NIST	National Institute of Standards and Technology
NTU	Nanyang Technological University
OCR	Optical Character Recognition
PCA	Principal Component Analysis
PCC	First Capital Command
R-CNN	Region-based Convolutional Neural Networks
ReLU	Rectified Linear Units
RNN	Recurrent Neural Network
ROI	Region Of Interest
RPN	Region Proposal Network
SIFT	Scale Invariant Feature Transform
SVM	Support Vector Machines
YOLO	You Only Look Once

CONTENTS

1	INTRODUCTION	14
1.1	MOTIVATION	14
1.2	OBJECTIVES	16
1.2.1	General Objectives	16
1.2.2	Specific Objectives	17
1.3	ORGANIZATION	17
2	LITERATURE REVIEW	18
2.1	METHODS IN COMPUTER VISION AND IMAGE RECOGNITION	18
2.1.1	Computer Vision	18
2.1.2	Deep Learning	20
2.1.3	Transfer Learning	23
2.1.4	Data Augmentation	25
2.1.5	Object Location	27
2.2	TATTOO RECOGNITION	30
2.2.1	Tattoo Recognition Roadmap	30
2.2.2	Tattoo Detection	33
2.2.3	Tattoo Location	36
2.2.4	Tattoo Classification	38
3	DATA AND METHODS	41
3.1	TATTOO DETECTION	41
3.1.1	Methods	41
3.1.2	Datasets for Tattoo Detection	42
3.2	TATTOO LOCATION	44
3.2.1	Methods	44
3.2.2	Datasets for Tattoo Location	45
3.3	TATTOO CLASSIFICATION	47
3.3.1	Methods	47
3.3.2	Datasets for Tattoo Classification	48
4	EXPERIMENTS AND RESULTS	51
4.1	TATTOO DETECTION	51
4.1.1	Evaluation of Feature Extractors	51
4.1.2	Evaluation of the Effect of Data Augmentation	52
4.1.3	Comparison with Other Dataset	53
4.1.4	Qualitative Analysis	55
4.2	TATTOO LOCATION	56
4.2.1	Evaluation of the Mask R-CNN for Tattoo Location	57
4.2.2	Evaluation of Mask R-CNN and Fine-Tuning	58
4.2.3	Qualitative Analysis	60
4.3	TATTOO CLASSIFICATION	62
4.3.1	Baseline Experiments	62
4.3.2	Evaluation of the Effect of Data Augmentation	64
4.3.3	Qualitative Analysis	65

5	CONCLUSIONS AND FUTURE WORKS	72
	REFERENCES	75
	APPENDIX	85
	APPENDIX A – TATTOO LOCATION - MASK R-CNN FINE TUNING	86
	APPENDIX B – MASK R-CNN DEFAULT PARAMETERS	91
	APPENDIX C – TATTOO CLASSIFICATION CONFUSION MATRIX	92
	APPENDIX D – RESULTS OF UNKNOWN CLASSES CLASSIFICA- TION	95

1 INTRODUCTION

1.1 MOTIVATION

Tattoos are widely used around the world, representing both a form of human expression and a kind of art. For more than 5,000 years (JAIN *et al.*, 2007) they have been used and can bring a lot of information about people, including their stories, their personality traits, groups they belong to, among others (ROHITH *et al.*, 2020). In their almost unique traits, tattoos may carry information that goes beyond the reasons that led people to make it, which can also be classified as a kind of biometrics and, consequently, they can be useful in identifying its carriers.

The term biometrics is described as the science of recognizing an individual based on their physical or behavioral traits (JAIN *et al.*, 2006). It is also described as the automated use of physiological or behavioral characteristics to determine or verify an individual's identity (SHARMA, 2015). Currently, biometrics are useful for many applications such as cell phone unblocking, bank transaction authentication, access control and especially those related to security, including people monitoring, criminal identification, identity certification and border control (LABATI *et al.*, 2016).

Biometrics is divided into two major areas: hard biometrics and soft biometrics (ABDELWHAB; VIRIRI, 2018). On the one hand, the former is characterized by being unique, distinctive and permanent, and includes face (ZHAO *et al.*, 2003; MARSICO *et al.*, 2014; IBSEN *et al.*, 2021), fingerprint (BORRA *et al.*, 2016; SONI; MAHESH, 2018), iris (NITHYA; LAKSHMI, 2015; NGUYEN *et al.*, 2017; RATTANI; DERAKHSHANI, 2017), palmprint (CHEN *et al.*, 2010; ZHANG *et al.*, 2012; ALI *et al.*, 2018), voice (FARRÚS, 2018), ear shape (ABAZA *et al.*, 2013), brain biometrics (GUI *et al.*, 2019), DNA (ZAHID *et al.*, 2019; KATSANIS *et al.*, 2022), among others (National Research Council, 2010; DUBAL; BHARADI, 2016; DONG *et al.*, 2017). On the other hand, soft biometrics are those that are not deterministic or permanent, designed to address less well-defined biometric traits, such as skin blemishes, scars, marks, ethnicity, age, gender (GUO; MU, 2014), gait (VERLEKAR *et al.*, 2018), signature (SERDOUK *et al.*, 2017; KAUR; JINDAL, 2016) and tattoos (JAIN *et al.*, 2004; DANTCHEVA *et al.*, 2010; UNAR *et al.*, 2014).

In the biometrics field, tattoo recognition becomes an important theme of research, given its complexity and applicability. Unar *et al.* (2014) established some criteria that help to qualify

biometrics in terms of Universality, Distinctiveness, Invariance, Collectability, Performance and Acceptability. Following this perspective, in Table 1 is possible to have a comparison between tattoo recognition and other biometrics regarding those criteria, in a comprehensive way.

Table 1 – Comparison between some common biometrics features and tattoos (✓ means “Yes”, ✗ means “No”, and “–” means “Does not apply”).

	Face	Fingerprint	Palmprint	Iris	DNA	Voice	Tattoos
Do all people have it?	✓	✓	✓	✓	✓	✗	✗
Is it discriminating in people?	✓	✓	✓	✓	✓	✗	✗
Is it found in a well-defined location?	✓	✓	✓	✓	–	–	✗
Is it invariant over time?	✓	✓	✓	✓	✓	✗	✗
Does it have well-defined characteristics?	✓	✓	✓	✓	✓	✓	✗
Does it have a pattern?	✓	✓	✓	✓	✓	✓	✗
Is it easy to collect?	✓	✓	✗	✗	✗	✓	✓
Are there well-established technologies for its recognition?	✓	✓	✓	✓	✓	✓	✗
Is the available technology assertive?	✓	✓	✓	✓	✓	✓	✗

Source: own author.

Compared to other biometrics, tattoos bring with them a series of characteristics that make them very difficult to recognize. Other biometrics usually have well-defined standards, robust techniques and well-established methods for their treatment and recognition, standardized data capture and storage, and other factors that help its reliability and robustness. However, tattoos still do not have such characteristics and requirements.

As a matter of fact, conventional biometrics (face, fingerprint, and iris, for instance) have well-established methods for preprocessing and re-identification based on its patterns. However, the same does not hold for tattoo images, when used as biometrics, which makes tattoo recognition a unique problem, since a tattoo is not standardized in shape, size, color, location, position, style, symmetry, stroke or fill.

In addition, the effective use of almost all the above-mentioned biometrics for real-world applications rely on computer vision methods. Such as methods are necessary for processing raw images as well as identifying/re-identifying individuals according to their biometric traits.

Apart from the issues related to processing and using general biometrics, tattoo recognition has a singular complexity because it can be divided into several sub-problems, each one equally important, that includes identification, location, segmentation, classification, re-identification, de-identification, among others.

Because of the wide scope of the topic, the published studies are fragmented, usually focusing on very specific issues. That is, the authors choose to present a method to solve some specific problem related to tattoo recognition. However, it is still missing a thorough review of

the different problems and proposed solutions.

On the other hand, from the point of view of the applicability of the theme, tattoo recognition has a great application in many areas, especially in law enforcement, where there is a great lack of methods and applications to be used in practice to recognize people by using tattoo information.

For this purpose, tattoos can provide an important source of biometric information which has great utility (ACTON; ROSSI, 2008; HARBERT, 2015) because tattoos can be used for identifying not only suspects but, also, victims (LEE *et al.*, 2008; FANG *et al.*, 2018). In addition, the subject has raised studies on ethical and social issues that may encompass the topic (BACCHINI; LORUSSO, 2017).

Tattoos can also be used both to recognize its bearer and, also, in some situations identify some personal characteristics, such as information about gangs and past facts. In Brazil, for example, some gangs often tattoo specific images to identify their members, or also are used to identify specific facts, like a clown, tattooed by people who had killed a police man, or specific numbers to determine to which gang they are associated.

Thus, the development of new methods and studies about tattoo recognition can bring a high contribution to this area of application. New image recognition methods have been developed, including deep learning based, and their application to tattoo recognition problems can contribute significantly to this research area.

1.2 OBJECTIVES

1.2.1 General Objectives

Tattoo recognition carries a large roadmap of problems to be solved, basically divided into preprocessing problems and recognition problems. Preprocessing problems includes tattoo detection, location, and segmentation. Identification problems include tattoo classification, re-identification and de-identification. Although many methods have already been presented to address these issues, this area is still opened to be studied, since no one has achieved results good enough to be considered completely solved.

Considering the complexity of the tattoo recognition roadmap (Section 2.2.1), and the importance to improve this research field based on the more recent image recognition approaches, and facing the open area to research in this field, the aim of this study **is to propose new**

approaches to tattoo detection, location, and classification problems.

1.2.2 Specific Objectives

In order to evaluate the capability to recognize images with tattoo, the present study propose a new group of tattoo datasets and approaches to three specific problems in tattoo recognition.

1. Tattoo detection: presents a method to determine whether an image contains a tattoo or not.
2. Tattoo location: presents a method to determine where are the tattoos in an image, inserting a bounding box around de tattoo area.
3. Open world tattoo classification: presents a method to determine a label to a tattoo based on a group of known labels and, if the image does not belong to any class, determine it as unknown.

1.3 ORGANIZATION

The presented work is divided in five chapters, as follows: the second chapter presents a complete review of the literature related with the problems and techniques addressed in this study in tattoo recognition, including a review of the methods, environment, and results that have been achieved so far to each problem. The third chapter presents the proposed methodology, the datasets, and the environment used to perform this research. Chapter four present the results achieved and a discussion about them. Finally, chapter five presents the conclusions and future works related with the problems, its challenges and suggestions for possible directions for further research.

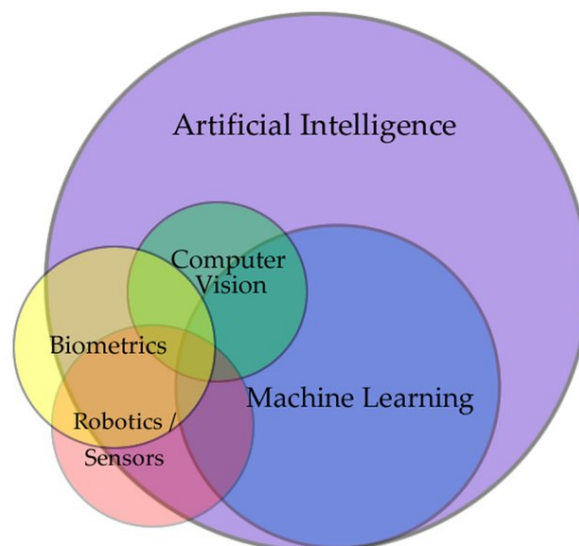
2 LITERATURE REVIEW

2.1 METHODS IN COMPUTER VISION AND IMAGE RECOGNITION

2.1.1 Computer Vision

Computer vision (CV) is a field of Artificial Intelligence (AI) that focus on understanding the content of digital images (Figure 1). “Typically, this involves developing methods that attempt to reproduce the capability of human vision” (BROWNLEE, 2019, access in April 21st, 2022).

Figure 1 – Computer Vision in Artificial Intelligence field.



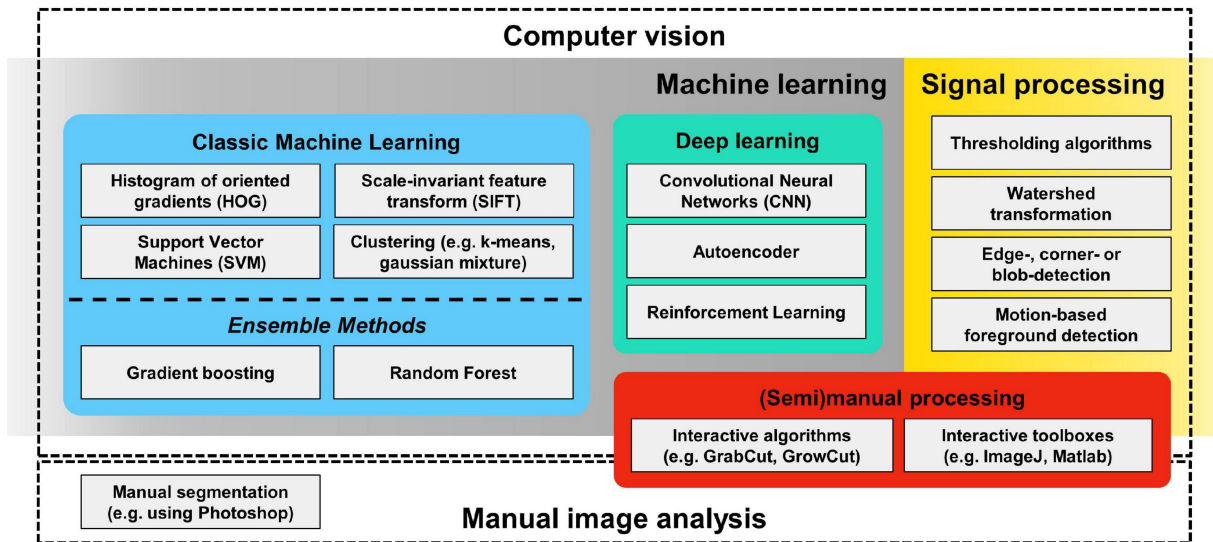
Source: Viejo *et al.* (2019).

In addition, Szeliski (2022) cites that computer vision “is an interdisciplinary field at the intersection of signal processing and machine learning, which is concerned with the automatic and semiautomatic extraction of information from digital images” (Figure 2).

To Mihajlovic (2019, access in April 21st, 2022), computer vision is “the field of computer science that focuses on replicating parts of the complexity of the human vision system and enabling computers to identify and process objects in images and videos in the same way that humans do”.

Brownlee (2019, access in April 21st, 2022) defines computer vision as “a field of study that seeks to develop techniques to help computers “see” and understand the content of digital

Figure 2 – Approaches to computer vision methods.



Source: Lurig *et al.* (2021).

images such as photographs and videos”.

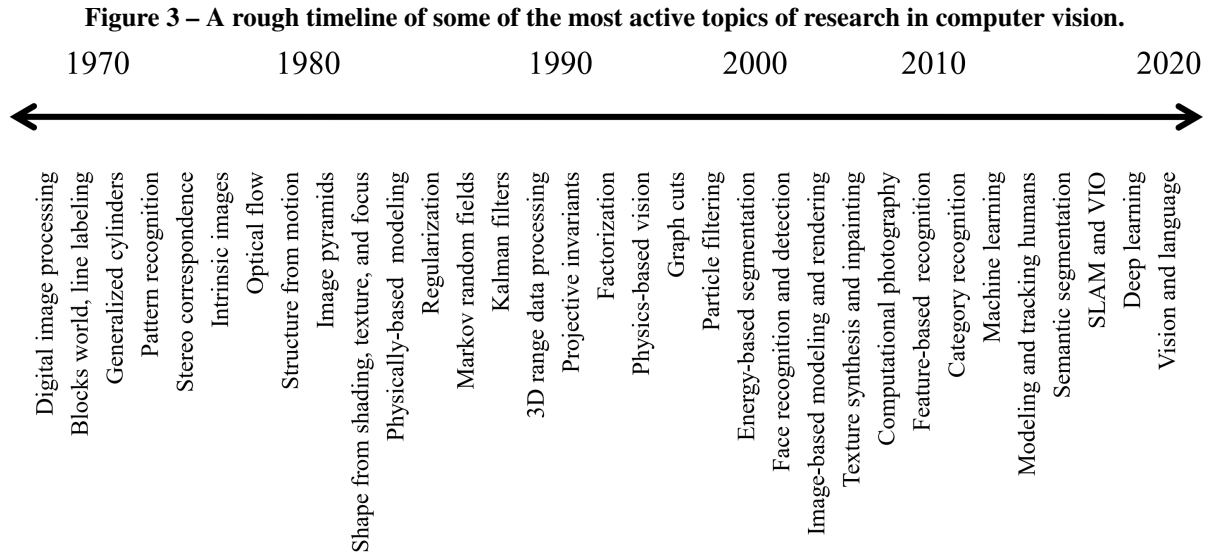
Computer vision covers many applications, that includes object classification, object identification, object verification, object detection, image segmentation, object recognition, video motion analysis, image restoration, scene reconstruction, object de-identification, among others.

In application areas, it is used in surveillance, healthcare, security, biometrics, robotics, industry, self-driving cars, Optical Character Recognition (OCR), machine inspection, retail (e.g., automated checkout), motion capture, engineering, agriculture, microbiology, ecology, among many others.

Concerning the research in computer vision, Szeliski (2022) presents a rough timeline of some of the most active topics in the last years (Figure 3).

Regarding to the methods applied in computer vision, Lurig *et al.* (2021) divide its evolution in four waves. In the first wave (Hand-Crafted Features), researchers in the 1970s and 1980s “developed different ways to perform feature extraction from raw pixel data. Such features tended to be low-level features, such as lines, edges, texture or lighting, but provided us with the initial basic geometric understanding of the data contained in images”. The methods developed includes watershed algorithm, Canny and Sobel filters, Hough transforms, Principal Component Analysis (PCA), Scale Invariant Feature Transform (SIFT) and Histogram of Oriented Gradients (HOG).

In the second wave (Initial Machine-Learning Approaches), “it became apparent that without image standardization, those low- and intermediate-level features will often fall short of



Source: Szeliski (2022).

producing sufficiently robust CV algorithms” (LURIG *et al.*, 2021). So, machine-learning starts acquired more generalizable applications using unsupervised algorithms to attempt to identify previously unidentified patterns on unlabeled data, such as k-means and Gaussian Mixture Models (GMM). However, it is in the supervised domain that machine learning for CV has been most successful by using Support Vector Machines (SVM) in image classification, recognition, among others.

The third wave (Ensemble Methods) brings the ensemble methods, that represent a “slightly different philosophical approach to machine learning, in which multiple models are trained to solve the same task and their individual results are combined to obtain an even better model performance” (LURIG *et al.*, 2021), and are divided in two main families: bagging and boosting.

Finally, the fourth wave (Deep Learning) brings the most actual approaches and the state-of-the-art in computer vision, the deep learning based methods. “Deep learning refers to a family of machine learning methods based on hierarchical artificial neural networks, most notably, CNN” (LURIG *et al.*, 2021).

2.1.2 Deep Learning

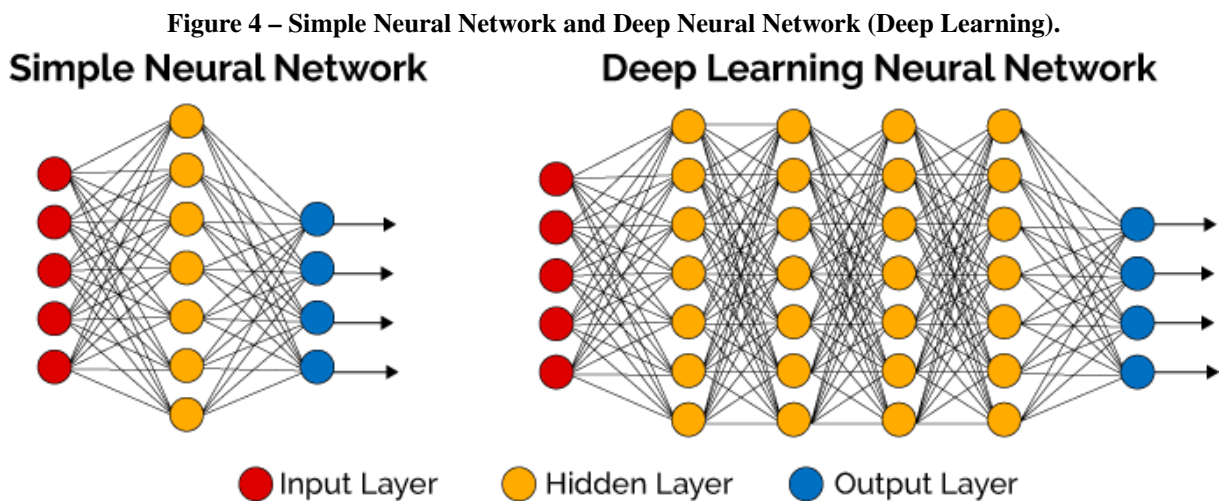
Deep Learning is a class of machine learning methods based on learning data representations. According to Hao *et al.* (2016), “deep learning is a branch of machine learning that tries to model high-level abstractions of data using multiple layers of neurons consisting of complex

structures or non-linear transformations”.

To LeCun *et al.* (2015), deep learning methods are “representation-learning methods with multiple levels of representation, obtained by composing simple but non-linear modules that each transform the representation at one level (starting with the raw input) into a representation at a higher, slightly more abstract level”.

In a more simple way, Goodfellow *et al.* (2016) describes deep learning as an AI field responsible to solve “tasks that are easy for people to perform but hard for people to describe formally — problems that we solve intuitively, that feel automatic, like recognizing spoken words or faces in images”. Also, according to the author, deep learning models “is to allow computers to learn from experience and understand the world in terms of a hierarchy of concepts, with each concept defined through its relation to simpler concepts”, and because this hierarchy of concepts is deep, with many layers, “we call this approach to AI deep learning”.

While conventional neural networks are formed by simple layers, deep networks are formed by a higher number of layers (Figure 4).



Source: Academy (2022, access in April 21st, 2022).

Different from shallow or simple machine learning, “deep learning uses a cascade of layers of nonlinear processing units for feature extraction and transformation. It allows computers to learn from a hierarchical representation of the data where higher level features are derived from lower level features” (HAO *et al.*, 2016).

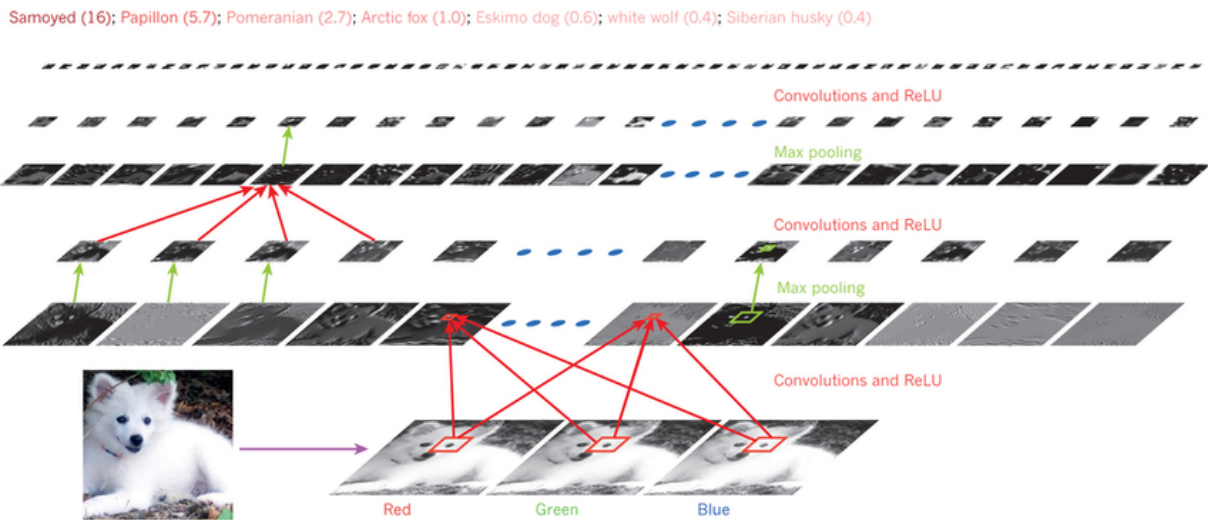
Deep learning introduced the concept of end-to-end learning, where the machine is just given a dataset of information and a deep learning model is trained on the given data to discover the underlying patterns in classes (O’MAHONY *et al.*, 2019).

There are a huge number of variants of deep architectures, and different architectures can

be used to represent different data sources. For example, CNN is the most popular architecture for image recognition, and Recurrent Neural Network (RNN) is more applicable to sequential tasks such as handwriting or speech recognition (HAO *et al.*, 2016).

CNN is structured as a series of stages (Figure 5). Units in a convolutional layer are organized in feature maps, within which each unit is connected to local patches in the feature maps of the previous layer through a set of weights called a filter bank (LECUN *et al.*, 2015).

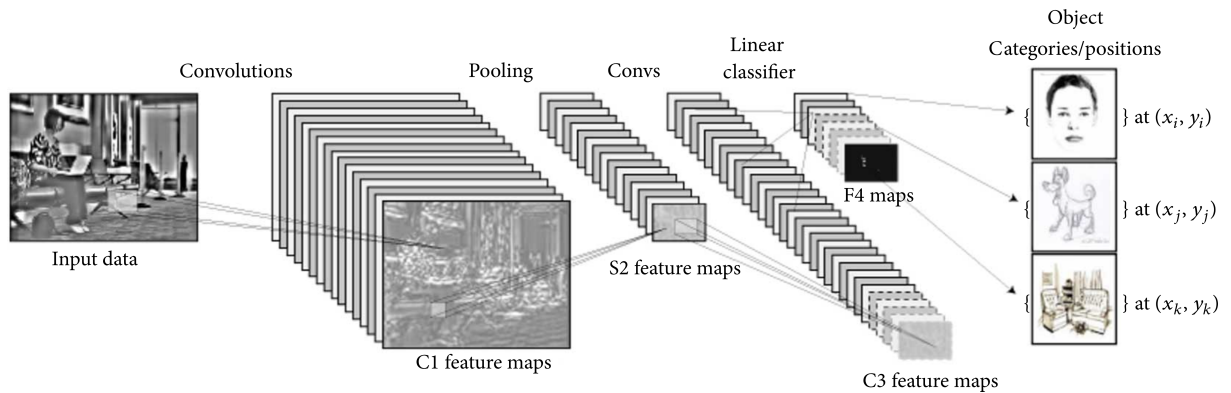
Figure 5 – Inside a convolutional network.



Source: LeCun *et al.* (2015).

“A CNN comprises three main types of neural layers, namely, (i) convolutional layers, (ii) pooling layers, and (iii) fully connected layers. Each type of layer plays a different role” (VOULODIMOS *et al.*, 2018) (Figure 6).

Figure 6 – Example architecture of a CNN for a computer vision task (object detection).



Source: Voulodimos *et al.* (2018).

As the deep learning in general, CNN also have a huge number of popular architectures, currently used as base network to feature extractor, for example. These networks include AlexNet

(KRIZHEVSKY *et al.*, 2017), ZF Net (ZEILER; FERGUS, 2013), VGG Net (SIMONYAN; ZISSERMAN, 2014), GoogLeNet (SZEGEDY *et al.*, 2015a), Microsoft ResNet (HE *et al.*, 2015), R-CNN (GIRSHICK *et al.*, 2014), Fast R-CNN (GIRSHICK, 2015) and Faster R-CNN (REN *et al.*, 2017)), Generative Adversarial Networks (GOODFELLOW *et al.*, 2014), Generating Image Descriptions (KARPATHY; FEI-FEI, 2014) and Spatial Transformer Networks (JADERBERG *et al.*, 2015).

2.1.3 Transfer Learning

Bozinovski (2020) defines transfer learning as “a machine learning method where a learning model developed for a first learning task is reused as the starting point for a learning model in a second learning task”.

To Zhuang *et al.* (2019), transfer learning “aims at improving the performance of target learners on target domains by transferring the knowledge contained in different but related source domains”.

Complementary, Bozinovski (2020) cites that “it is a research problem in machine learning that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem”.

In other words, the concept back to transfer learning is the same of human learning, where a person who plays piano very well, for example, have the capability to transfer his/her knowledge to another person who does not plays piano (WEISS *et al.*, 2016). In this machine learning case, a network previously trained in a specific scenario should transfer some information to a new network, inheriting important information from that trained network.

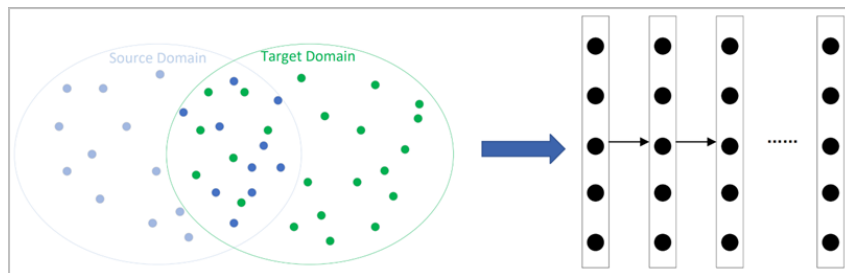
Following Tan *et al.* (2018), transfer learning methods can be divided in four main categories:

- Instances-based: “refers to use a specific weight adjustment strategy, select partial instances from the source domain as supplements to the training set in the target domain by assigning appropriate weight values to these selected instances” (Figure 7).
- Mapping-based: “refers to mapping instances from the source domain and target domain into a new data space” (Figure 8).
- Network-based: “refers to the reuse of the partial network that pre-trained in the source

domain, including its network structure and connection parameters, to transfer it to be a part of deep neural network which used in target domain” (Figure 9).

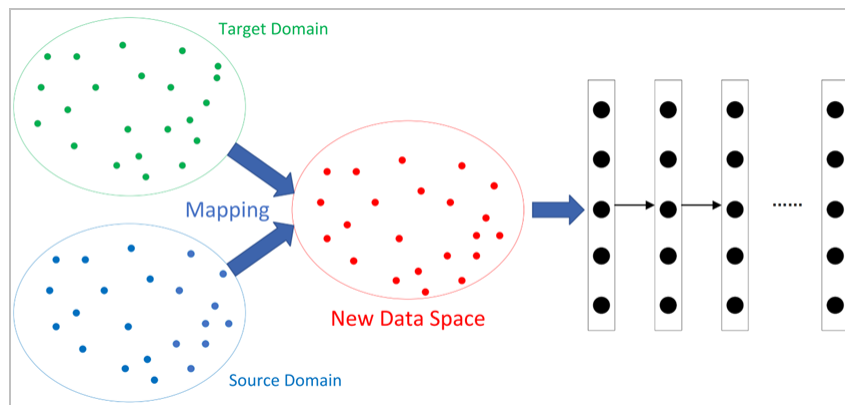
- Adversarial-based: “refers to introduce adversarial technology inspired by GAN to find transferable representations that are applicable to both the source domain and the target domain” (Figure 10).

Figure 7 – Sketch map of instances-based deep transfer learning.



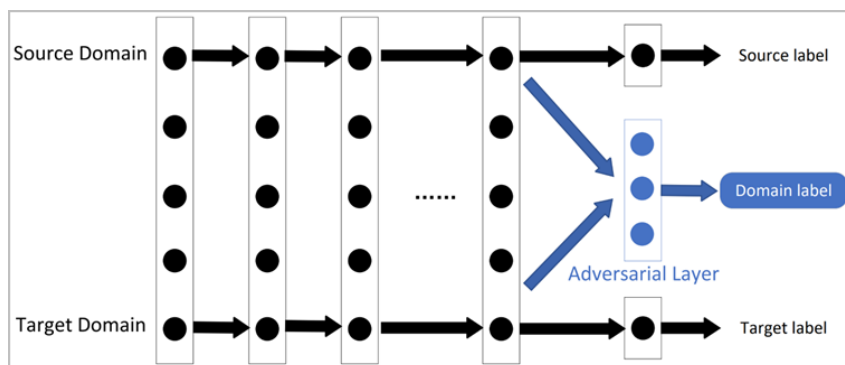
Source: Tan *et al.* (2018).

Figure 8 – Sketch map of mapping-based deep transfer learning.



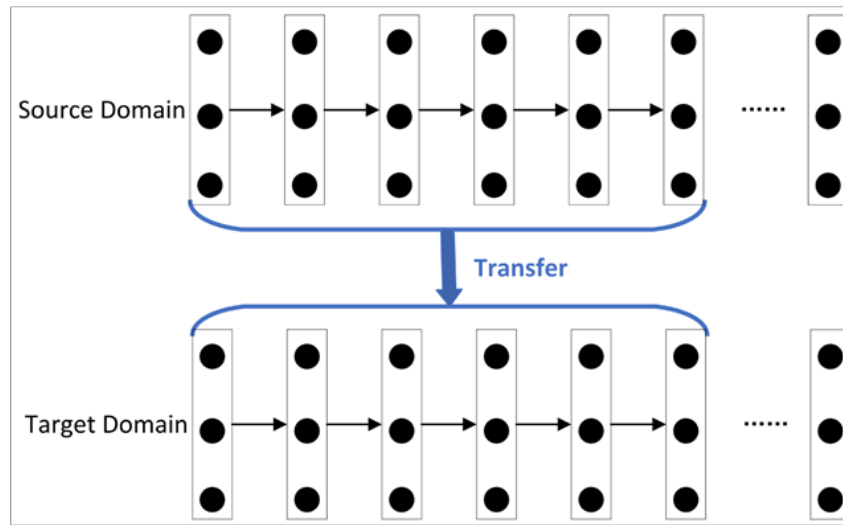
Source: Tan *et al.* (2018).

Figure 9 – Sketch map of network-based deep transfer learning.



Source: Tan *et al.* (2018).

Figure 10 – Sketch map of adversarial-based deep transfer learning.



Source: Tan *et al.* (2018).

In models where images are used, as CNN's, for example, transfer learning allows starting with the learned features on known datasets such as ImageNet (DENG *et al.*, 2009), FaceNet (SCHROFF *et al.*, 2015), MS COCO (LIN *et al.*, 2014), Google Open Images Dataset (KUZNETSOVA *et al.*, 2020), among others, and adjust these features and perhaps the structure of the model to suit the new dataset/task instead of starting the learning process on the data from scratch with random weight initialization (HUSSAIN *et al.*, 2018).

2.1.4 Data Augmentation

Data is the main raw material in deep learning environments, specially when models are applied to problems involving images. In some cases, obtain sufficient data to build robust models and avoid overfitting, for example, is a great challenge (PEREZ; WANG, 2017).

As a way of circumventing these restrictions and barriers, data augmentation brings a way to adding slightly modified copies of already existing data or a set of synthetic data obtained from the original ones.

Shorten and Khoshgoftaar (2019) defines data augmentation as “a suite of techniques that enhance the size and quality of training datasets such that better Deep Learning models can be built using them”. The authors also presented a quick taxonomy of the data augmentations methods (Figure 11).

Also, according Mikolajczyk and Grochowski (2018) and Shorten and Khoshgoftaar (2019), there are a number of methods that can be applied to augment image data, which can be

Figure 11 – A taxonomy of image data augmentation.



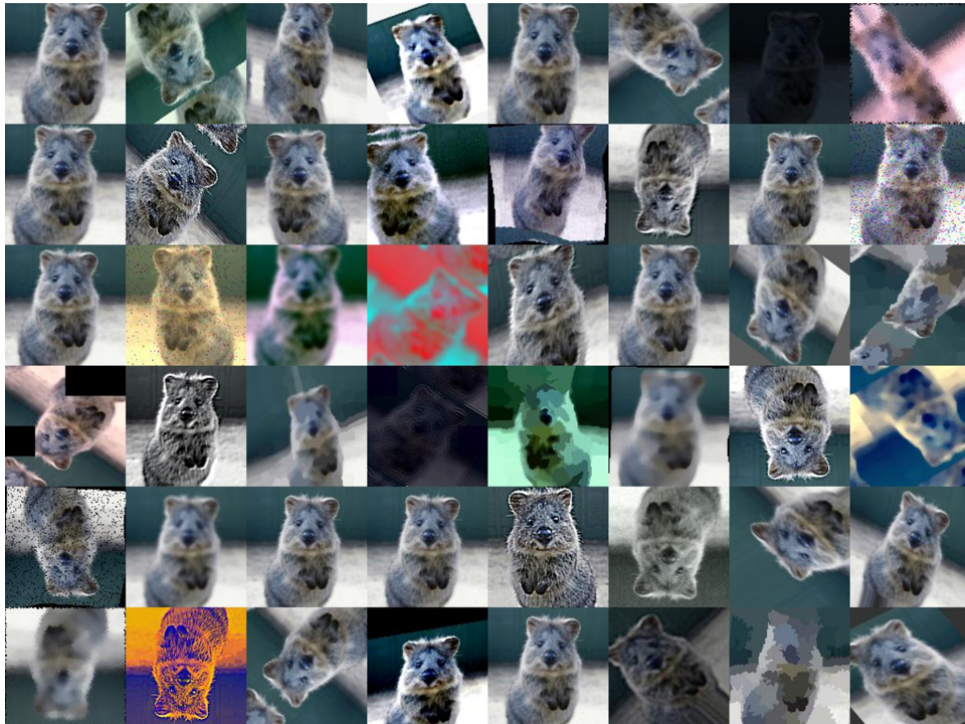
Source: Shorten and Khoshgoftaar (2019).

divided into a few categories:

- **Traditional transformations:** perform combination of the affine image transformations and color modification such as rotation, translation, flipping, cropping, reflection, scaling (zoom in/out) and shearing, geometric distortions or deformations such as histogram equalization, enhancing contrast or brightness, white-balancing, sharpening, noise injection and blurring.
- **Generative Adversarial Networks:** GANs are found to be extremely useful in many image generation and manipulation problems like text-to-image synthesis, super-resolution (generating high-resolution image out of low-resolution one), image-to-image translation (e.g. convert sketches to image), image blending (mixing selected parts of two images to get a new one), image inpainting (restoring missing pieces of an image).
- **Texture transfer:** the goal of the texture transfer is to synthesize a texture from a texture-source image while constraining the semantic content of a content-source image.

Figure 12 presents a group of images generated by traditional data augmentation methods, and Figure 13 presents a group of images generated by Cycle GANs data augmentation methods.

Figure 12 – Data augmentation samples.



Source: Shorten and Khoshgoftaar (2019).

Figure 13 – Data augmentation samples with Cycle GANs.



Source: Shorten and Khoshgoftaar (2019).

2.1.5 Object Location

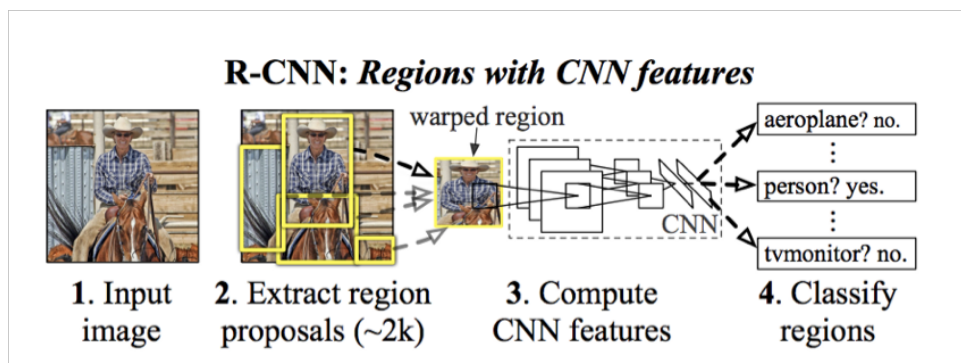
An important class of problem in computer vision is related with object detection, i.e., given an image, identify objects in it, localizing the position of each item instance, drawing a bounding box around de object.

Currently, deep learning are the most used approaches for this problem with the state-of-

the-art in this area based on the evolution of the R-CNN networks, there are R-CNN (GIRSHICK *et al.*, 2014), Fast R-CNN (GIRSHICK, 2015), Faster R-CNN + RPN (REN *et al.*, 2017) and Mask R-CNN (HE *et al.*, 2017), and YOLO networks (REDMON *et al.*, 2015; REDMON; FARHADI, 2016; REDMON; FARHADI, 2018; BOCHKOVSKIY *et al.*, 2020), currently in version 5.

Girshick *et al.* (2014) elaborates one of the first breakthroughs of the use of CNNs in an object detection system called the R-CNN network that had a much higher object detection performance than other popular methods at the time. The model is divided in three modules. The first generates category-independent region proposals, defining the set of candidate detections available to the detector. The second module is a large convolutional neural network that extracts a fixed-length feature vector from each region. The third module is a set of class specific linear SVMs (Figure 14).

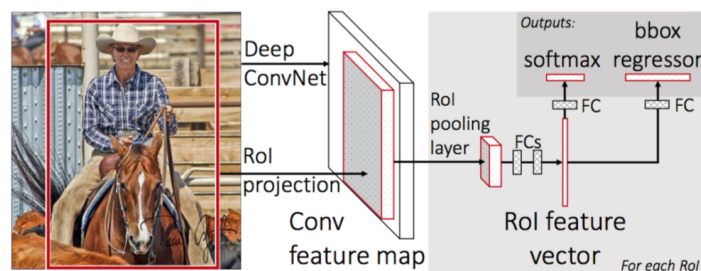
Figure 14 – R-CNN architecture.



Source: Girshick *et al.* (2014).

Girshick (2015) employs several innovations to improve training and testing speed in the original R-CNN network, while also increasing detection accuracy by training the very deep VGG16 network, which was 9× faster than R-CNN and also more accurate (Figure 15).

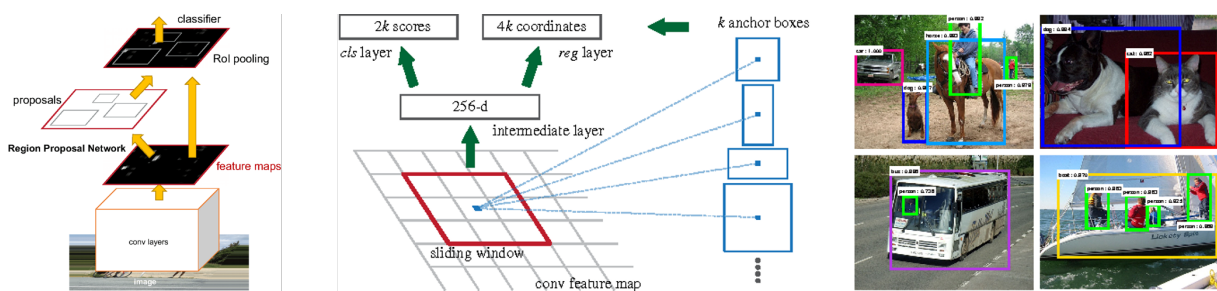
Figure 15 – Fast R-CNN architecture.



Source: Girshick (2015).

Ren *et al.* (2017) introduced a RPN that “shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals”. According to the authors, “an RPN is a fully convolutional network that simultaneously predicts object bounds and objectness scores at each position”. Also, “the RPN is trained end-to-end to generate high-quality region proposals, which are used by Fast R-CNN for detection”. As a result, they presented a network resulted by the merge RPN and Fast R-CNN into a single network called Faster R-CNN (Figure 16).

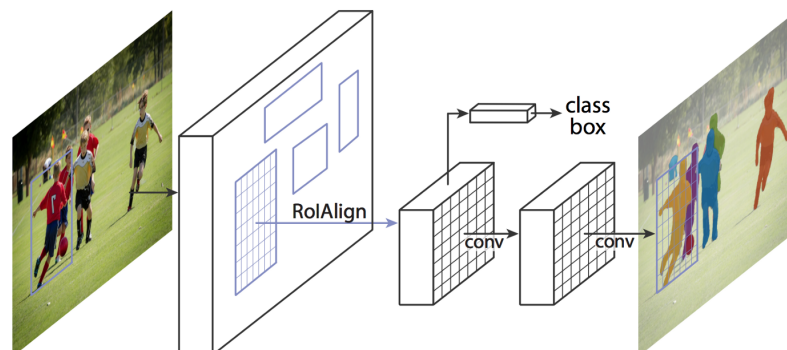
Figure 16 – Faster R-CNN architecture. Left: Faster R-CNN is a single, unified network for object detection. Center: RPN. Right: Example detections using RPN proposals on PASCAL VOC 2007 dataset test.



Source: Ren *et al.* (2017).

After that, He *et al.* (2017) presented the Mask R-CNN, “a conceptually simple, flexible, and general framework for object instance segmentation”. This network detects objects in an image while simultaneously generating a high-quality segmentation mask for each instance. Also, the network “extends FasterR-CNN by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition” (Figure 17).

Figure 17 – Mask R-CNN framework.

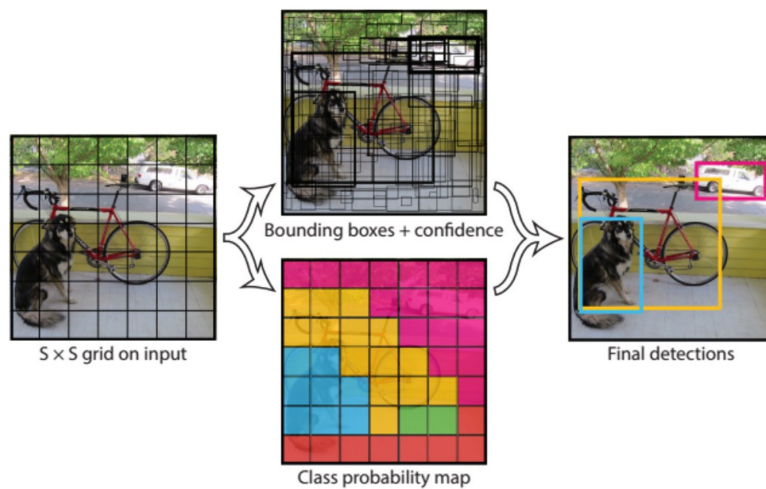
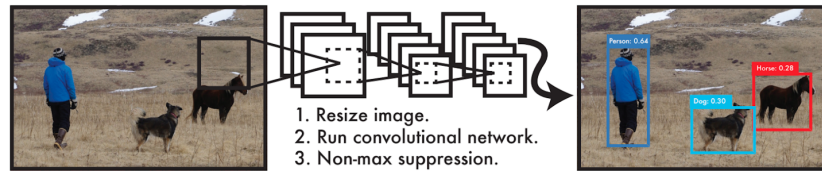


Source: He *et al.* (2017).

In Redmon *et al.* (2015), the authors “frame object detection as a regression problem to spatially separated bounding boxes and associated class probabilities”. Then, “a single neural

network predicts bounding boxes and class probabilities directly from full images in one evaluation. Since the whole detection pipeline is a single network, it can be optimized end-to-end directly on detection performance” (Figure 18).

Figure 18 – The YOLO Detection System.



Source: Redmon *et al.* (2015).

2.2 TATTOO RECOGNITION

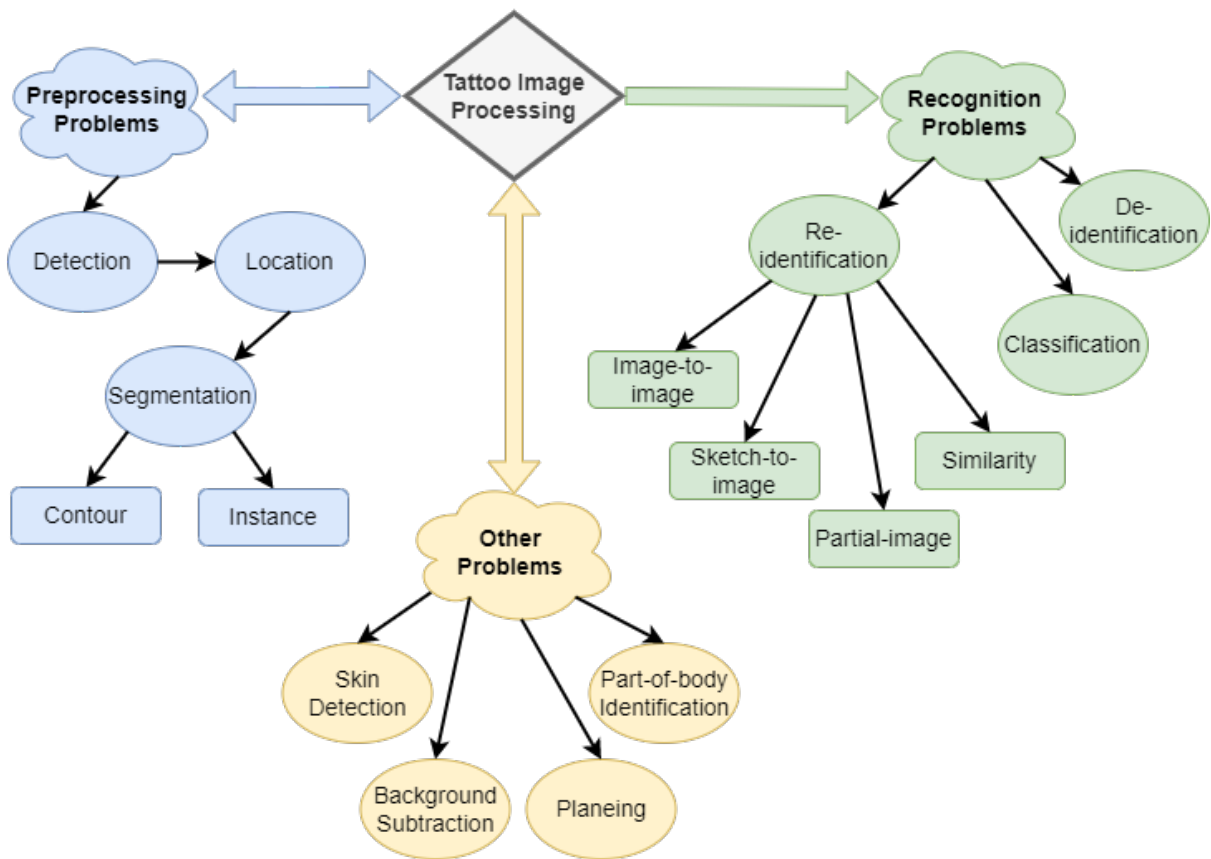
2.2.1 Tattoo Recognition Roadmap

The tattoo recognition roadmap comprehends a group of problems that covers the entire cycle to recognize a tattoo, from detecting it on an image to classifying or re-identifying a tattoo in a database. The source image can be an entire tattoo, a sketch or partial information, and the expected result can be a corresponding image, a group of similar images, a class or simply boolean information, a bounding box, a cropped or erased image. Because of that, tattoo recognition roadmap requires a specific methodology for each step or related problem.

Basically, tattoo recognition roadmap is divided into three main groups of problems: pre-processing, recognition and other (Figure 19).

The pre-processing step is not related to any recognition process itself, but rather to

Figure 19 – Tattoo recognition roadmap.



Source: own author.

prepare the original image for the recognition step. It includes the following problems:

- Detection: determines whether an image (of a human) has a tattoo or not.
- Location: finds where in the image the tattoo is found, and returns a bounding-box around the corresponding tattoo region.
- Contour Segmentation: crops out the exact contour of the tattoo from the rest of the image, separating the tattoo area from the surrounding background.
- Semantic Segmentation: crops further the tattoo image, separating each distinct object represented in the tattoo.

The pre-processing tasks have the challenge to prepare the image to be submitted to the next step, receiving a raw image and returning an image without the noise represented by the background. This process is as important as the recognition process because the better the pre-processing, the more efficient the recognition.

The recognition processing, which includes the methods that effectively perform the tattoo recognition, addresses the following problems:

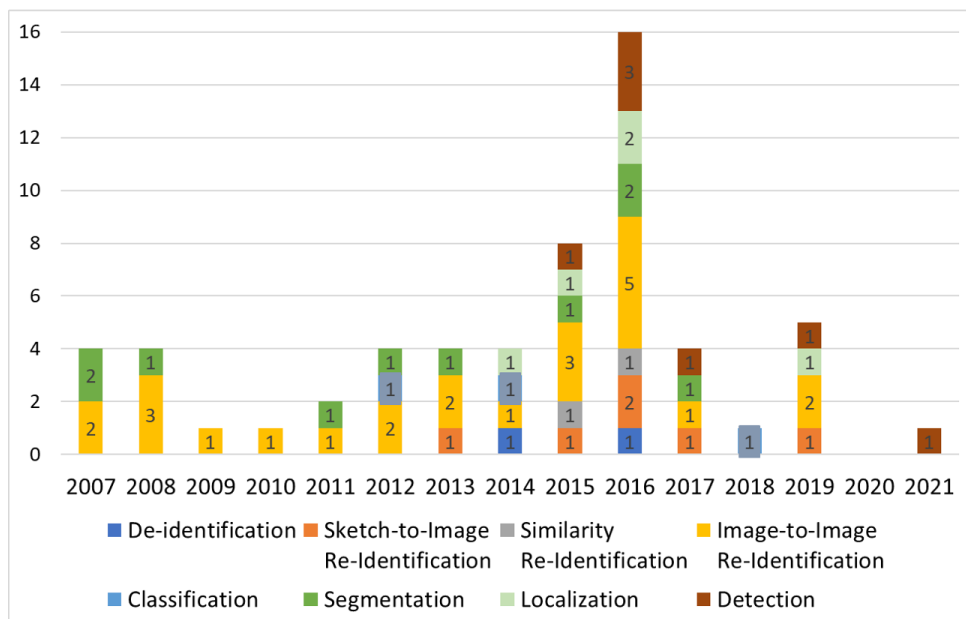
- Re-identification: searches for an image in a dataset of tattoo images, returning the one that is the most similar to the searched tattoo. This process can be further divided into other specific models:
 - Image-to-Image: consists of, given a sample tattoo image, finding the most similar image in a database.
 - Sketch-to-Image: consists of, given a hand-drawn sketch of a tattoo, finding the most similar image in a database which contains photos of real tattoos.
 - Similar Groups: consists of, given a sample tattoo image, finding a group of tattoos that are similar to the one searched and that have the same pattern, but not necessarily the original image searched.
 - Partial Image-to-Image: consists of, given a partially occluded tattoo image, finding images that have the searched image as part of a complete image in the database.
- Classification: given a tattoo image, give a description for the object or objects that make up that image, returning labels to the input image.
- De-Identification: consist of a process of erasing the tattoo from an image, a process also known as anonymization.

Other problems that can help to improve the quality of the results in tattoo recognition, include, but are not limited to, the following issues:

- Skin Detection: consists of, given an image, detecting regions that represent a person skin.
- Background Subtraction: consists of, given an image, subtracts all parts that do not represent a person or parts of a person.
- Planeing: corrects an image in which the tattoo is in irregular areas, making it flat.
- Part-of-body Identification: consists of, given an image, determines which body parts are present in the image.

Many of these tattoo problems have been studied in the last years, and some of them have no published studies so far has this research gone. Figure 20 presents the number of publication by theme over the years. This information was collected by an exhaustive search on the main portals of scientific publications, including IEEEXplore, Scopus, Google Scholar, Springer, ACM, Elsevier, and others, during all the development of this research. There were considered all publications directly applied to some tattoo recognition theme. The problems not presented in the figure indicates that no publication was found for that specific problem applied to tattoos.

Figure 20 – Number of relevant publications by category and year.



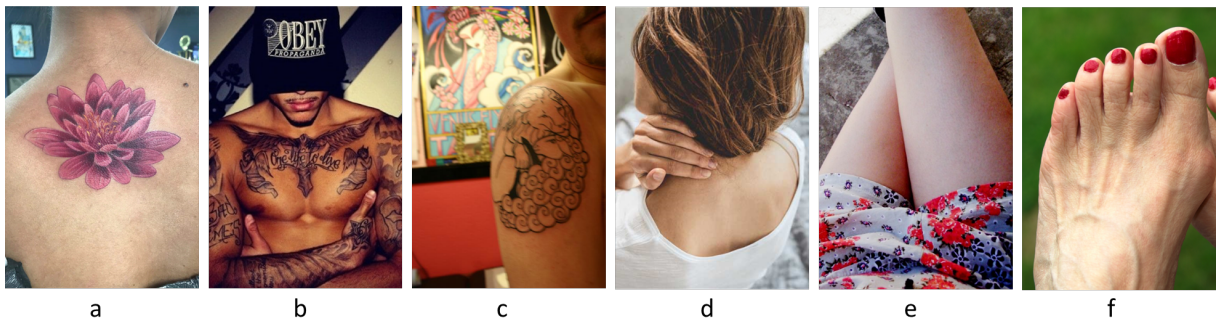
Source: own author.

2.2.2 Tattoo Detection

The tattoo detection problem consists in determining whether an image contains a tattoo or not. Figure 21 shows some examples of images of people with and without tattoos. Despite the theoretical simplicity of the concept, the detection process is not such a simple task at all, as there are no defined standards for what a tattoo is in terms of patterns of shape, color, size, proportion to the individual and, mainly, its location on the body. In addition, a single image can have several tattoos. Furthermore, the image background or colored clothes can introduce significant noise to the detection process, since its complexity may be confused with tattoos. Some of these issues are shown in the images shown in Figure 21, for both tattoo and non-tattoo

images.

Figure 21 – Example of images, used for the detection task, of people with and without tattoos. Image "a" is an example of a well-behaved situation, where the tattoo is a well-defined, and the surrounding background is a clear skin. The Image "b" there are multiple tattoos, and a bounding-box of them will include other areas of the body where there is no tattoo. Also, the letters of the hat may disturb the recognition process. In image "c" the painting in the backward has similar patterns to the tattoo, and they are close to each other. The image "d" is a good example of non-tattoo, easily identified. In the image "e", the design of the clothes in contrast with the skin may be confused as a tattoo. In image "f" painted nails may be confused as a tattoo.



Source: own author.

Tattoo detection plays a fundamental role in the initial image filtering and data selection, and its importance has been neglected for many years by the scientific community Xu *et al.* (2016), and the first publication on this topic came up only more recently, in 2015 Ngan and Grother (2015).

After that seminal publication, several other studies followed. Table 2 presents a summary of the results recently published for the tattoo detection problem. In this table, column “Best Result 1” refers to results obtained when using the same dataset for both, training the model and testing the model, evaluating its accuracy. In the column “Best Result 2”, the accuracy presented was achieved by testing the model using a different dataset from that used to train the model (this issue will be addressed later in this work).

Table 2 – Tattoo Detection Published Results.

Reference	Year	Method	Best Result 1	Best Result 2
Ngan <i>et al.</i> (2016)	2015	not cited	96.30% acc.	-
Xu <i>et al.</i> (2016)	2016	CNN	98.80% acc.	93.78%
Di and Patel (2016)	2016	AlexNet + 2-Class SVM	99.83% acc.	-
Di and Patel (2017)	2017	AlexNet + 2-Class SVM	99.83% acc.	-
Xu and Kong (2016)	2016	Decision tree	52.38% acc.	-
Sun <i>et al.</i> (2016)	2016	Faster R-CNN	98.25% acc.	80.66%
Han <i>et al.</i> (2019)	2019	Faster R-CNN	87.10% recall (WebTattoo) 61.70% recall (Tatt-C)	80.00%

Source: own author.

In Table 2 it is noticed a diversity of machine learning methods as well as seemingly good results. However, training and testing over the same dataset may lead to biased results, since generalization capability of the classifiers are not evaluated. This fact raises an important issue, the datasets used. Table 3 describes the databases used in each study. Due to the large diversity of tattoo types and the lack of standards for capturing images, datasets can be very different from each other. As a consequence, it is difficult or even unreasonable to compare results.

Table 3 – Tattoo datasets referenced in the tattoo detection bibliography.

Reference	Dataset Name	Tattoo Images	Non-Tattoo Images
Ngan <i>et al.</i> (2016)	Tatt-C	1,349	1,000
Xu <i>et al.</i> (2016)	Tatt-C	1,349	1,000
	NTU_Flickr	5,740	4,260
Di and Patel (2016)	Tatt-C	1,349	1,000
Di and Patel (2017)	Tatt-C	1,349	1,000
Xu and Kong (2016)	Unidentified	547	-
	Tatt-C	3,839	-
Sun <i>et al.</i> (2016)	PASCAL Visual Object Classes (VOC) 2007	-	9,963
	NTU_Flickr	5,740	4,260
Han <i>et al.</i> (2019)	Tatt-C	1,349	1,000
	NTU_Flickr	5,740	4,260

Source: own author.

The first published results appeared in response to the challenge published by National Institute of Standards and Technology (NIST) (NGAN; GROTHOR, 2015). In this scenario, four institutions presented results, with the company MorphoTrek presenting the best performance Ngan *et al.* (2016). Unfortunately, the algorithms used by the participants of the NIST competition were not published. This fact turned out impossible to carry out external validation tests, which was criticized in Xu *et al.* (2016).

Based on this scenario, Xu *et al.* (2016) suggested evaluating whether the dataset available in Ngan and Grother (2015) and Ngan *et al.* (2016) was sufficiently comprehensive to ensure that the results presented could be generalized. Therefore, the authors presented a CNN trained in two scenarios: the Tatt-C dataset (NGAN; GROTHOR, 2015) and a dataset obtained from Flickr (NTU_Flickr). The experiment consisted of training the network with one of the datasets and validating with the other, and vice-versa. Initially, in the same scenario presented in Ngan *et al.* (2016), the CNN described in Xu *et al.* (2016) had a slightly higher performance, increasing the previous accuracy. In subsequent tests, networks trained on the NIST dataset and validated on the NTU_Flickr dataset performed worse than the other way around. Finally, the authors showed that as the training dataset increases, the result accuracy also improves.

In Di and Patel (2016) and Di and Patel (2017), the authors also propose using a CNN for tattoo detection, also basing their study on the Tatt-C dataset. The proposed model consists in extracting features through fine-tuning the AlexNet network and, then, applying a linear SVM to determine whether an image has tattoos or not. The proposed algorithm was also compared with Ngan *et al.* (2016), and obtained an improvement of 3.2% compared to the best initial result.

Another approach based on decision trees was presented by Xu and Kong (2016), this time in its own dataset, with less expressive results.

In Sun *et al.* (2016) the authors present a deep learning region-based method to tattoo detection, the Faster R-CNN, which is based on a fine-tuning of the VGG_CNN_M_1024 network. The training data was also based on Tatt-C dataset, but now joining other 9,963 images without tattoos divided in 20 object categories from the PASCAL Visual Object Classes (VOC) 2007 dataset Everingham *et al.* (2010). Their results were also compared with those presented in Ngan *et al.* (2016), and its performance had an accuracy 1.95% better than that presented by MorphoTrek in Ngan *et al.* (2016).

More recently, Han *et al.* (2019) presented a detection model also using a Faster R-CNN. In this model, the detection problem was classified as an instance of the image recovery system, where learning and detection were performed simultaneously. The authors also present a result based on the recall percentage, which was compared with the results obtained in Sun *et al.* (2016). While Sun *et al.* (2016) presented a recall of 45% to 0.1 FPPI (false positive per image) for the Tatt-C dataset, the authors in Han *et al.* (2019) presented a result of 61.7% for the same dataset and 87.1% to a dataset obtained from the internet (called WebTattoo). In summary, it is difficult to compare different works due to differences in test procedures, metrics and the datasets used.

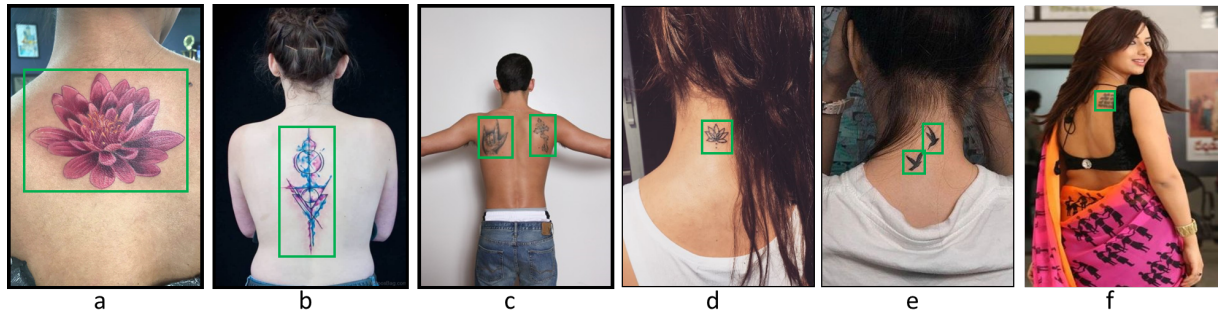
2.2.3 Tattoo Location

The next step, after detecting a tattoo, is the tattoo location. This issue is essential for the next steps of tattoo image processing, since it is responsible for removing most of the image that is not useful, such as clothes and background.

Unlike the detection, which is a binary decision problem, tattoo location requires the return of a set of coordinates that allows identifying the specific regions where tattoos are found in an image. In other words, the challenge is to locate where one or more tattoos are found in the original image, and enclose them into rectangular bounding boxes (Figure 22).

Usually, the bounding box created around a tattoo is defined by a set of coordinates

Figure 22 – Examples of tattoo location.



Source: own author.

(x, y, w, h) , such that x and y represent the initial coordinates of the bounding box, and w and h represent the width and height of the frame, respectively (SUN *et al.*, 2016). The objective is to find, for each disjoint tattoo in the image, a frame with the minimal dimensions capable of enclosing all the tattoo (KIM *et al.*, 2016).

Table 4 shows a summary of the papers found in the literature, including methods and results for each study. Although the first publications did not present numerical results, the most recent studies took important contributions to the tattoo location problem using machine learning methods.

Table 4 – Papers approaching the tattoo location problem.

Reference	Method	Best Result	Dataset Size
Marcetić <i>et al.</i> (2014)	Decision rules in RGB color space + morphological operations	4.9% False Positive Rate	204
Ngan <i>et al.</i> (2016)	not shown	97% acc.	16,716
Kim <i>et al.</i> (2016)	Center-surround feature location	66.20% acc. @Tatt-C	4,308
Kim <i>et al.</i> (2016)	Graph-cut	70.46% acc. @Tatt-C	4,308
Sun <i>et al.</i> (2016)	Faster R-CNN	98.25% acc.	23,802
Han <i>et al.</i> (2019)	Faster R-CNN	Tatt-C: 61:7% recall@0.1FPPI WebTattoo: 87:1% recall@0.1FPPI	8,026

Source: own author.

Marcetić *et al.* (2014) proposed a two-step approach for locating tattoos in an image. First, the skin is detected by applying decision rules in the RGB color space, followed by geometric restrictions to eliminate skin-like color regions that do not belong to body parts. Next, potential tattoo regions are located in the cropped regions with a different skin color, obtained by the morphological operation of closure.

The location problem was also part of the NIST challenge mentioned before. The best accuracy of 97% was achieved by the MorphoTrek company. However, algorithms, data used and how the accuracy was measured were not made available for further validation tests.

Later, Kim *et al.* (2016) presented a specific study for tattoo location with two specific methods. The first is a center-surround feature location method. The second method used a graph-cut segmentation based on the image edges, a skin color model and a visual bump map. To evaluate the results, images were previously cropped manually. The second method achieved better accuracy than the first one, reaching 70.46% and 69.91% with the Tatt-C and Evil Tattoo datasets, respectively.

Sun *et al.* (2016) customized a Faster R-CNN, by fine-tuning the VGG_CNN_M_1024 network. The model was trained with Tatt-C and PASCAL Visual Object Classes (VOC) datasets with manually annotated bounding boxes around the tattoos. A validation step was done using the Nanyang Technological University (NTU) dataset with 10,000 images. The best accuracy was 98.25%, which was marginally higher (1.25%) than the results presented by Ngan *et al.* (2016), which was used as basis for comparison.

Another work made by Han *et al.* (2019) also used the Faster R-CNN model, and achieved 99% of accuracy when the method was applied to the Tatt-C dataset, in the same joint training process presented in their tattoo detection model.

2.2.4 Tattoo Classification

The tattoo classification problem consists of, given a tattoo image, classifying its semantic contents according to pre-established categories, such as person, animal, cat, dog, car, flag, etc. (Figure 23). In other words, the challenge is to automatically “translate” a tattoo into pre-established labels.

Some attempts to establish formal categories were presented by McCabe and Newton (2007), Wing (2011), and cited in some publications Jain *et al.* (2007), Lee *et al.* (2012), Jain *et al.* (2012), Marcetić *et al.* (2014), but ended up being questioned, since tattoos have many more categories than those listed in the proposed standards, which contained only 8 classes divided into 70 sub-classes. In Lee *et al.* (2012), for instance, manual annotations reached 1,737 different classes, after analyzing 64,000 images.

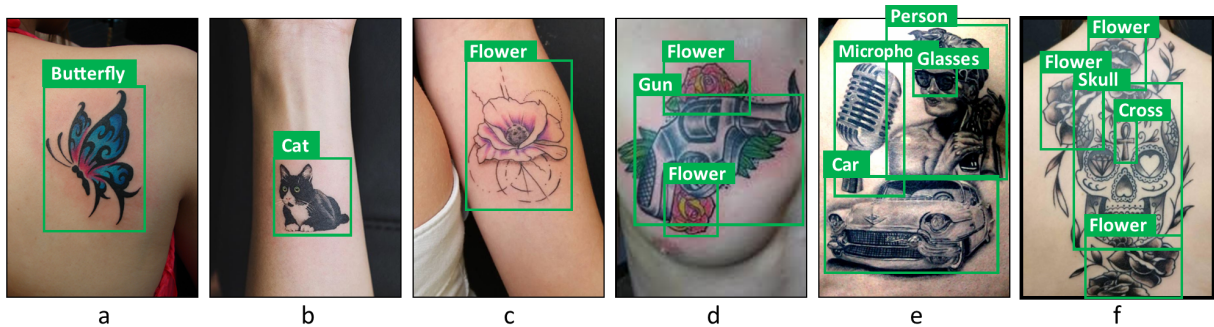
In other publications, such as Jain *et al.* (2009), Lee *et al.* (2012), classes were used only as pre-filter annotations on images as a way of decreasing the dataset breadth used re-identifying tattoos problem.

In this way, the classifying tattoos problem is reported as a great complexity, since it is about images without any pre-selected patterns, and it can still very often have multiple classes

in the same image, or even be an abstract image, without a specific category. Because it is an artistic work, there is no standard in what is tattooed on a person.

Perhaps because of this complexity, only a few studies have been published addressing this topic, even though it has great practical applicability (Table 5). Also, the studies published have been based in small number of classes and the datasets used are also small.

Figure 23 – Example of tattoos to be classified. Items “a”, “b” and “c” shows images with one label, like a butterfly, a cat and a flower. Images “d”, “e” and “f” shows tattoos with multiple labels, like a gun and flowers; a microphone, a person, a car and sunglasses; a skull, flowers, a heart, a diamond and a cross.



Source: own author.

Table 5 – Tattoo Classification Published Results.

Reference	Methods	Classes	Best Result	Dataset Size
Heflin <i>et al.</i> (2012)	1-class SVM	15	many	18,922
Wilber <i>et al.</i> (2014)	SVM	5	63.8% acc.	1,200
Jiawang and Yuan (2018)	Triplet GAN	-	82% acc.	200

Source: own author.

In the study of Heflin *et al.* (2012), authors, after locating and segmenting the tattoos, apply 1-class SVM for classifying images. They used that method for training a classifier for each class, separately, and repeated the procedure for 15 different classes divided in three different classifiers: animals (6 classes), humans (3 classes) and miscellaneous (6 classes). The authors present a plot of positive classification rate versus negative classification rate to each group of classes as classification results, and a plot comparing the results achieved with an adaptation of the method used by Lee *et al.* (2012), although the objective of this last work was not classifying images, but re-identify them. Unfortunately, they did not present a specific numerical measure of accuracy to the classifiers.

Wilber *et al.* (2014) proposed a mid-level classification, where they try to balance the accuracy level with performance. They proposed a method called Exemplar Codes, based on linear classifiers with probabilistic normalization to balance classifiers. First, features are

extracted from tattoo images by using multi scale pyramid of HOG. Next, a SVM was used to create the Exemplar Codes. This is done by classifying each simple positive class image with a set of negative samples and, then, a set of exemplars is generated. In the next step, they used the Random Forest method to classify the tattoo images, using the Exemplar Codes. All experiments were carried out on 5 tattoos classes (butterfly, skull, flower, star and dragon), with 50 to 80 images per class, plus 400 images of negative classes.

More recently, Jiawang and Yuan (2018) presented a Triplet GAN model aiming at obtaining a high accuracy in image classification with few labels. The model was initially trained with the MNIST dataset and, then, applied to a dataset of 200 tattoo images obtained on the internet. The results were compared with tests performed with the k-means, Linear SVM, k-Nearest Neighbors (kNN) and Naive Bayes methods. Overall, the proposed model obtained accuracy between 77% and 82%, while the other methods varied between 54% and 72%.

At the end, to the best of the knowledge gathered so far, no model was found in the literature capable of addressing all the problems related to this roadmap at the same time. In general, each problem is studied separately, using different approaches, methods and datasets, by different authors.

3 DATA AND METHODS

This study proposes three different methods, each one responsible for a different problem, as presented before: tattoo detection, tattoo location and tattoo classification.

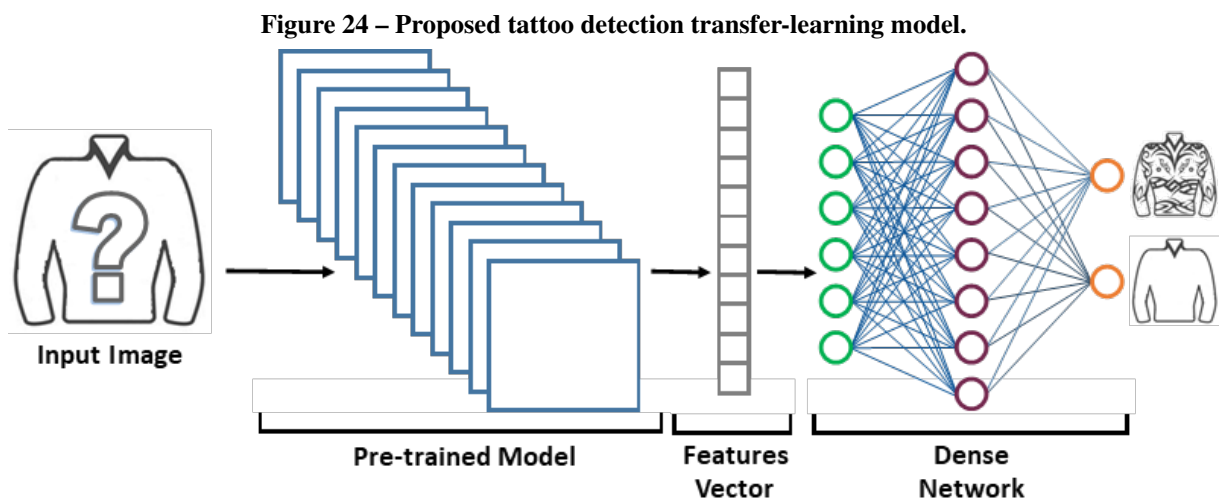
For each problem, specific datasets were proposed, each one having characteristics related to data type, volume, balance, among others, as will be shown in detail below. The datasets were also carefully elaborated, with the objective of creating a robust environment for the application of the methods and evaluation of results.

3.1 TATTOO DETECTION

3.1.1 Methods

The first problem addressed is the tattoo detection. In tattoo recognition roadmap, this method is important because it applies the first filter in a given image, determining whether the image contains or not a tattoo.

For this problem, a transfer learning approach was used, which has been shown excellent results in many classification problems, specially for image processing (ROMERO *et al.*, 2020b; GUTOSKI *et al.*, 2021; ROMERO *et al.*, 2020a). The basic idea is to use a CNN architecture, trained for a given problem, and re-use part of its architecture for another problem (Figure 24). Usually, this is accomplished by using the same type of data, in our case, images.



Source: own author.

The last layers of the Pre-trained Model are excluded and the remaining are re-used as a feature extractor. In other words, the knowledge learned by the trained network will be transferred to another similar problem. Therefore, using this procedure, a pre-trained network receives an image and provides a feature vector that represents that image in a high-dimensional embedded space (Features Vector).

The dense layer is composed by two hidden layers with 1,024 and 512 neuron, respectively, with ReLU activation function. Adam solver and categorical crossentropy loss function were used in this network.

In this working pipeline, the images of the training set are presented to the feature extractor and the output vector is forwarded to the input from a dense neural network, which is the trainable classifier for tattoo or non-tattoo classes by using a binary output layer.

3.1.2 Datasets for Tattoo Detection

Datasets are a fundamental part for all machine learning methods. In the one hand, choosing the correct dataset to perform a study is directly related to the quality of results. On the other hand, looking at the results without evaluating the dataset used can lead to misconceptions about the real quality of results. Considering that a dataset is a sample of the real world, it is particularly important for the efficiency of machine learning methods that the datasets used reflect the same diversity.

For multi-class datasets, the balance of samples in the classes is another important issue since, in general, classifiers are strongly biased towards the majority class. Unfortunately, many real-world datasets do not follow such principles. For instance, in the dataset Tatt-C, presented in Ngan and Grother (2015) and widely used in the literature, has images of the non-tattoo class predominantly of faces. Possibly, this can bias a classifier trained with this dataset, acquiring the misconception that images without tattoos are generally those with faces. This issue was criticized by Xu *et al.* (2016) in their publication. Despite this, that database was the most used, to date, for studies involving tattoos, although no longer available.

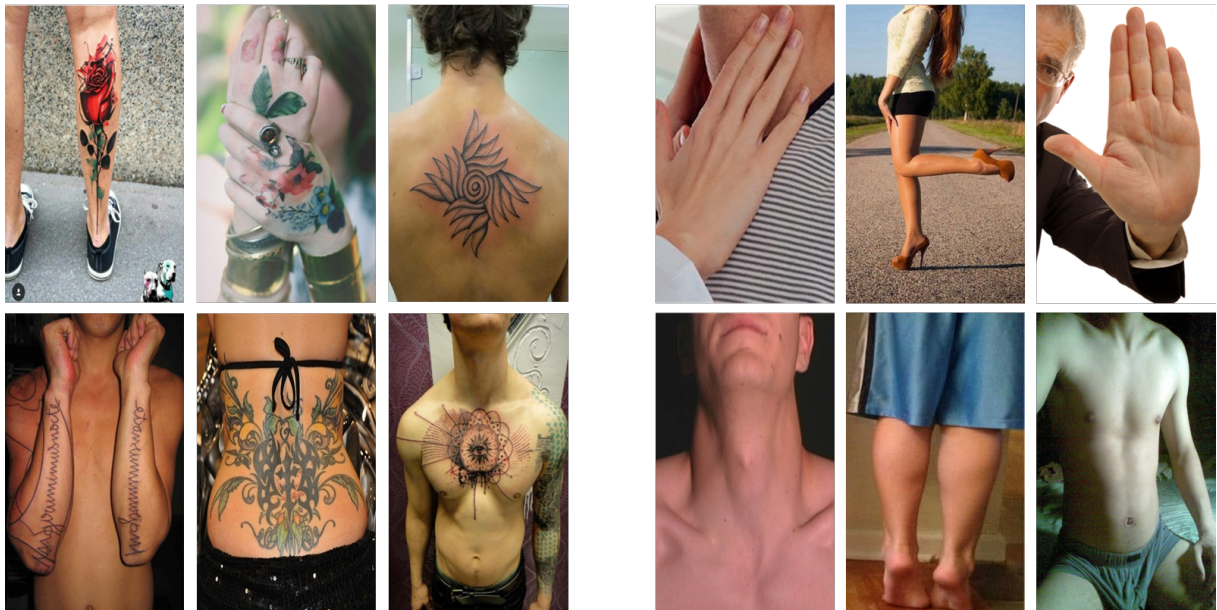
Considering the previous considerations, the datasets were designed for this specific study, taking care to maximize the image diversity and minimize possible biases in the results. Considering the need of two classes, namely, tattoo and non-tattoo, both must have diversity not only in terms of the specific part of the body that is in focus but, also, in the amount of background, distance to the tattoo within the image and framing pattern.

In addition, it is desirable to obtain data from different sources, to avoid possible bias due to the source of the information.

Two datasets were created, namely, TattDetectB and TattDetectF, with images extracted from internet at Bing¹ and Flickr², respectively. Each dataset was composed of 2,000 images of people, 1,000 for the tattoo and 1,000 for the non-tattoo class.

To obtain images for the proposed dataset, a web scraping technique was used. It consists of scanning internet pages, identifying images, and capturing. A Python script was used to perform web scraping. For each website, we performed the web scraping, searching for images with and without tattoos separately. Also, aiming at improving the diversity of images, we searched for images with tattoos combined with specific parts of the body, such as back, shoulder, arms, legs, etc. (see Figure 25). Such a procedure was done for both classes, tattoos and non-tattoos of TattDetectB and TattDetectF. Overall, this procedure helped to provide a good balance within the datasets.

Figure 25 – Samples of tattoo and non tattoo images in train datasets.



Source: own author.

¹ <http://bing.com>

² <http://flickr.com>

3.2 TATTOO LOCATION

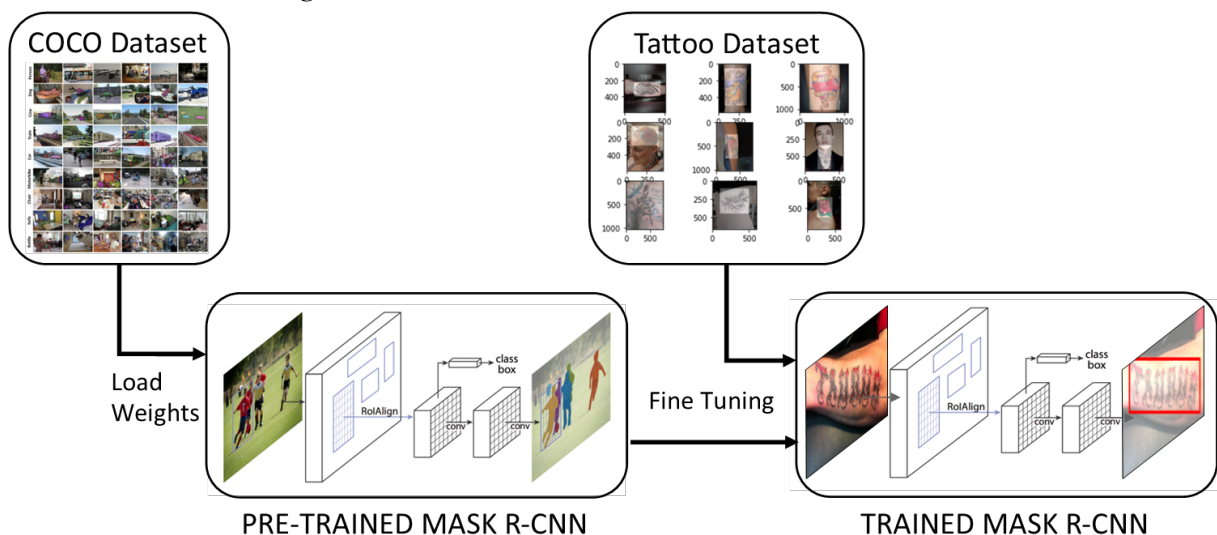
3.2.1 Methods

Tattoo location is the second problem related to the preprocessing group in tattoo recognition. This problem requires a method to find the exact tattoo location in a given image, returning a group of coordinates containing a position (top and left coordinates) and a size (width and height) representing the bounding box for each tattoo found.

For this problem, a Mask R-CNN network was used, which is one of the most recent approaches used for problems related to image location and segmentation. Although this network is capable of addressing those two problems at the same time, to this work only the tattoo location problem was approached.

As presented in Section 2.1.5, the Mask R-CNN contains an object detection layer which uses a RPN to locate objects and return a bounding box for each one. Therefore, the method used for tattoo location consists in initializing the Mask R-CNN with its original weights, and continue training the network by using a previously annotated tattoo dataset. Then, a fine-tuning in the network hyper-parameters is done to obtain a new set of hyper-parameters that improves the efficiency of the network for localizing tattoos in a given image (see Figure 26).

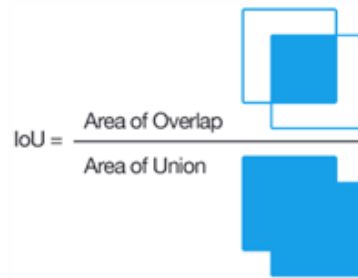
Figure 26 – Mask R-CNN for Tattoo Location Architecture.



Source: own author.

About the evaluation metrics, we used is the IoU (Figure 27) to compare a ground-truth bounding box versus a predicted bounding box. In terms of positive comparison, it is considered a positive pair if the IoU between the bounding boxes exceeds 0.7.

Figure 27 – Computing the IoU is as simple as dividing the area of overlap between the bounding boxes by the area of union.



Source: Rosebrock (2016, access in May 2nd, 2022).

For a global measure of quality for a dataset submitted to the network, the mean Average Precision (mAP) and standard deviation Average Precision (sdAP) metric were used, which computes the general average of the IoU of all images in the dataset, and the respective standard deviation. Using mAP it is possible to have an overview of how the network behaved for a specific dataset, for example, as a general measure for the training and validation datasets.

3.2.2 Datasets for Tattoo Location

The datasets used for tattoo location methods require special attention, since each image must be individually annotated with a specific bounding box information, i.e., the initial position (x,y) of the bounding box and its width and height (w,h) .

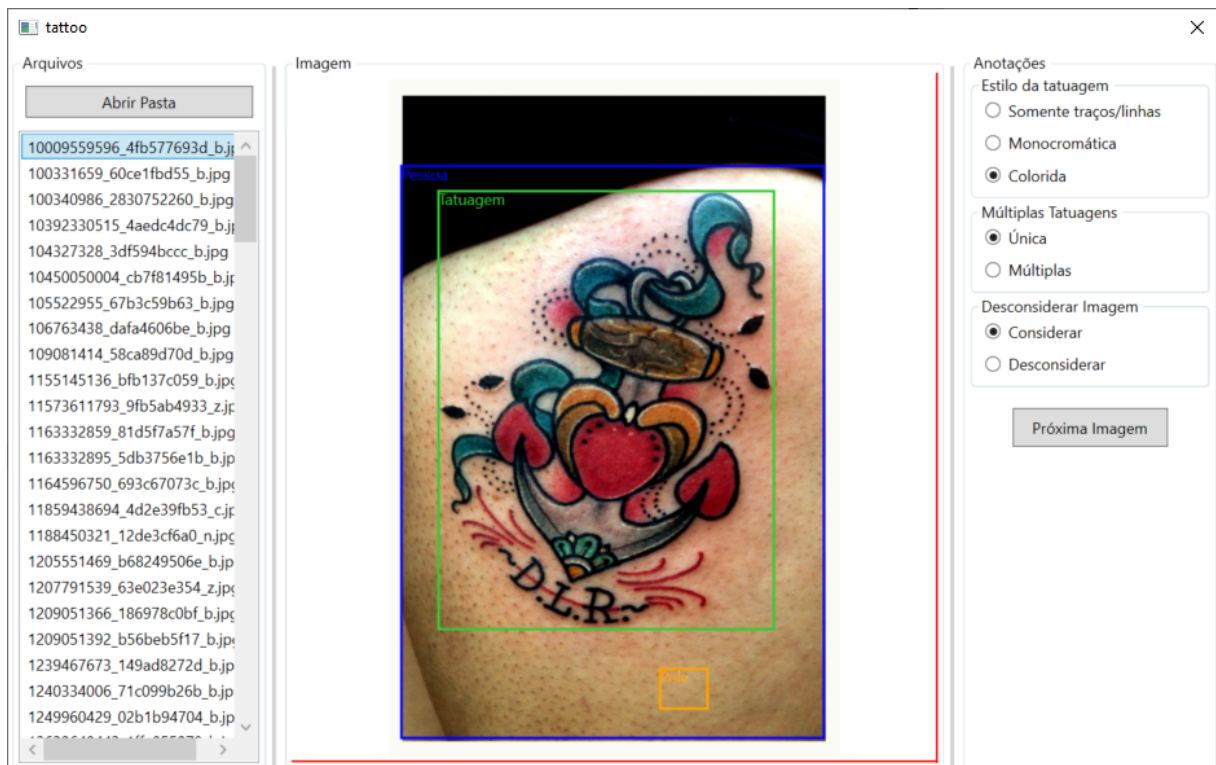
For this study, two datasets were created, one complementary to the other, containing 1,683 and 4,071 images, respectively, called TattLocA and TattLocB. Data were obtained from Bing and Flickr websites through web scraping, executed in Python. To obtain the data, several terms were used in order to obtain the largest possible variety of images in terms of size, style and colors.

Each image was manually annotated by using a system specially developed for this purpose, as shown in Figure 28. Annotations include the following information:

- Tattoo bounding box: coordinates and tattoo size.
- Person area: approximated area used by person on the image, used to calculate the proportions of person and tattoo in the image.
- Tattoo style: indicates whether tattoo style is color, monochromatic or outline.
- Multiple tattoo: indicates the existence of one or more tattoos in the image.

- Consider the image: indicates whether the image should be used or discarded. In general, very confusing or irregular images were disregarded.

Figure 28 – Tattoo annotation system.



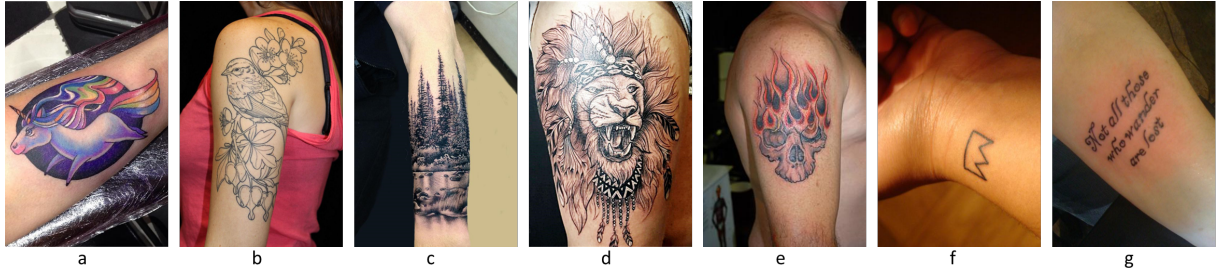
Source: own author.

After that, tattoos were classified in terms of style, color and proportion in the image (Figure 29). Color tattoos are those that uses more than one color to be filled. Outline tattoos are those unfilled, drawn only with lines. Monochromatic tattoos are those filled with only one color (usually black). There are also the text tattoos, represented by those with only text, in general, a particular case of outline tattoos.

In terms of size, tattoos were classified as small, medium and large. Small are those that occupy less than 1/3 of the total image area, medium are those that occupy more than 1/3 and less than 2/3 of the total image area, and large are those that occupy more than 2/3 of the total image area.

Accordingly to what was previously defined, the specifications of TattLocA and TattLocB datasets are as shown in Table 6 and Table 7, respectively.

Figure 29 – Sample of different tattoo styles and proportion. Image “a” represents a color tattoo, image “b” represents an outline tattoo, image “c” represents a monochromatic tattoo, image “d” represents a large size tattoo, image “e” represents a medium size tattoo, image “f” represents a small size tattoo and image “g” represents a textual tattoo.



Source: own author.

Table 6 – Dataset for Tattoo Location - TattLocA.

	Color			Monochromatic			Outline		
	Train	Validation	Total	Train	Validation	Total	Train	Validation	Total
Small	201	50	251	202	50	252	120	30	150
Medium	352	88	440	224	56	280	51	13	64
Large	123	31	154	62	16	78	11	3	14
Total	676	169	845	488	122	610	182	46	228

Source: own author.

Table 7 – Dataset for Tattoo Location - TattLocB.

	Color			Monochromatic			Outline		
	Train	Validation	Total	Train	Validation	Total	Train	Validation	Total
Small	490	122	612	487	122	609	458	115	573
Medium	650	162	812	446	112	558	308	77	385
Large	222	55	277	100	25	125	96	24	120
Total	862	216	1,078	1,034	258	1,292	1,361	340	1,701

Source: own author.

3.3 TATTOO CLASSIFICATION

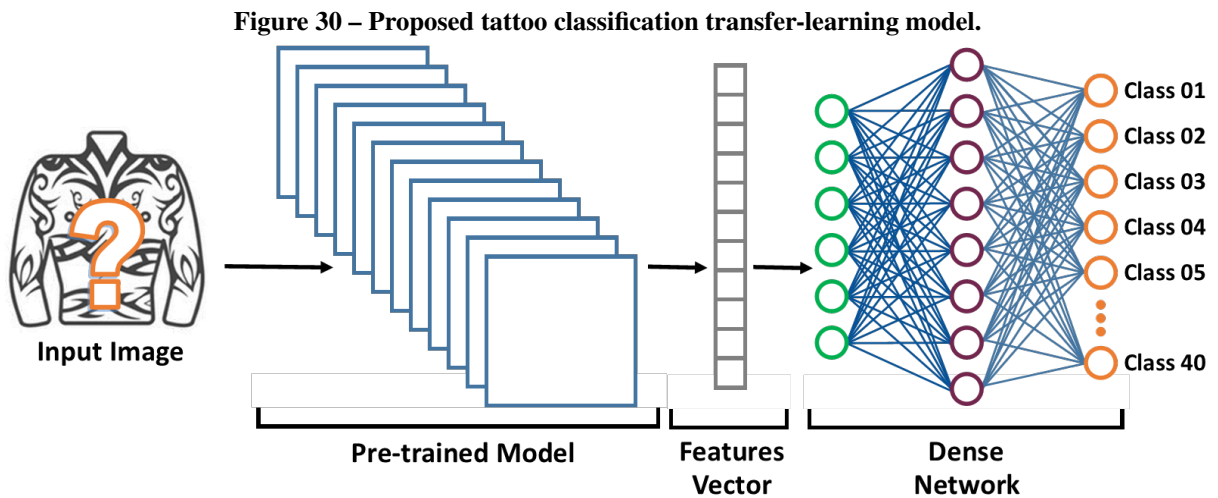
3.3.1 Methods

The last problem addressed in this work is the tattoo classification. Image classification has a great application in real world, and the development of methods that improve the capability of classifying objects is a great challenge.

Preliminary tests show that traditional methods, when applied to images with tattoos, do not give good results. Networks like YOLO (REDMON; FARHADI, 2016; LI; YANG, 2018; REDMON; FARHADI, 2018; BOCHKOVSKIY *et al.*, 2020), for instance, cannot satisfactorily classify tattoo images, most likely because they have not been trained with this sort of images.

Therefore, to tackle this problem, a transfer learning technique was also used. Similarly to the model used in tattoo detection (Section 3.1.1), the basic idea is, again, to use a CNN archi-

texture, trained for a given problem, and re-use part of this architecture for other problem, now classifying the contents of an image in specific classes, instead of a binary problem (Figure 30).



The dense layer is composed by two hidden layers with 1,024 and 512 neuron, respectively, with ReLU activation function. A total of 40 classes were selected to compose this classifier, explained later in Section 3.3.2. As in the tattoo detection problem were tested many feature extractors and the conclusion was that all of them achieved similar and good results, now the chosen feature extractor was the SqueezeNet, to use a network with a small number of features, unlike the first model. A final layer with softmax activator function was added with 40 outputs (classes). Adam solver and categorical crossentropy loss function were used in this network.

3.3.2 Datasets for Tattoo Classification

The dataset created for tattoo classification, named TattClass40, was composed by 40 classes with 100 images of each class, as follows: angel, bird, butterfly, car, cat, clock, clown, crucifix, demon, dog, dragon, eagle, eye, fish, flower, guitar, heart, horse, Jesus, joker, knife, leprechaun, lion, man face, octopus, owl, scorpion, skull, snake, spider, spider web, star, taz, tiger, weapon, wings, wizard, women face, yin-yang and zombie. Some image samples are shown in Figure 31.

These classes were chosen according to two main criteria: similarity between classes and its relevance for public security applications. For instance, angel, clown, demon, elf, Jesus,

Figure 31 – Sample of tattoo classes.



Source: own author.

joker, man face, wizard, women face and zombie have characteristics of human face. Classes such as bird, butterfly, dragon, eagle, owl and wings share, also, similar features, like wings.

Some classes have a special meaning to criminals, bringing a kind of encrypted communication, as these messages can hide the history of a crime or reveal a life of misdeeds. The following tattoos are examples of criminal meaning (MENEGHETTI, 2018, accessed in April 16th, 2022):

- Demon: “This is a tattoo usually done by someone who takes pleasure in killing. It is common to find the design of the devil in gunmen and hired killers, dangerous bandits who turn the act of taking the lives of other people (especially their rivals) into a craft”.
- Wizard: “Many inmates with a history of robberies on buses and commercial establishments have a wizard or a witch tattooed on their body. Some of these “criminal wizards” are specialists in weapons and explosives, with their favorite targets being stores with safes and ATMs”.
- Scorpion, Yin-yang and Numbers: “The design of a scorpion was the first tattoo used inside and outside São Paulo prisons to identify members of the First Capital Command (PCC), a well-known criminal faction originating in the state of São Paulo. Likewise, the

yin-yang tattooed on the body also indicates members of this organization. A tattoo with equivalent meaning of the PCC members is the number 1533 - a reference to the 15th letter of the alphabet (P) and the third one (C)".

- Leprechaun: "Criminals who tattoo a leprechaun on their body have a history of trouble, usually associated with drug abuse and trafficking. In the imprisoned population, the pixie tattoo also suggests a drug treatment activity in the past".
- Jesus: "On the arms, legs and chest, it is indicative of larceny (robbery followed by death). If located on the back, it could be a request for protection from the criminal".
- Clown: "If the tattoo has black tears, it means friends were killed by rivals. If the tears are red, the comrades were killed by the police. And if the clown cartoon is accompanied by skulls, it is almost certain that it is a police murderer. The number of skulls counts how many police officers were killed by the tattooed person".
- Spider: "In the criminal world, having a spider tattooed on the body means that one is a bandit who acts in a group".
- Taz: "The popular cartoon Taz is the best representation for criminals indicted for theft and robbery, and who act in a group".
- Octopus: "It is very common to find this animal tattooed on people accused of theft and burglary – usually bandits skilled in escaping".

The images in the dataset are already cropped around the tattoo image. As the goal is the tattoo classification, the images were prepared to have the lowest possible background, simulating as if all original images were passed through the tattoo location method, for example.

As shown before in Section 2.2.3, all published approaches for tattoo classification problem were applied in less diversified environments or with few samples and classes. So, the creation of high-quality datasets with more classes is essential for the development of robust models.

4 EXPERIMENTS AND RESULTS

4.1 TATTOO DETECTION

The experiments in tattoo detection were carried out using the Orange platform (DEM-SAR *et al.*, 2013) and complemented with scripts in the Python language, both running in the Windows 64 bits environment, on a desktop computer with Intel i7-8565U CPU @1.80GHz processor, and 16 GB of memory.

Based on model presented in Section 3.1.1, the main objectives in this experiment are:

1. Evaluate different feature extractors and check which ones can bring better results in tattoo detection.
2. Evaluate the effect of data augmentation on the classification results.
3. Compare results using different datasets.
4. Perform a qualitative analysis of the results.

4.1.1 Evaluation of Feature Extractors

In the first experiment, it was aimed at answering the following question: “Which feature extractor can lead to better detection results?”. The following architectures were tested as feature extractors: SqueezeNet (IANDOLA *et al.*, 2016), Inception-v3 (SZEGEDY *et al.*, 2015b), VGG-16 and VGG-19 (SIMONYAN; ZISSERMAN, 2015). For these Deep Neural Networks (DNNs), the length of the feature vectors are 1,000, 2,408, 4,096 and 4,096, respectively.

In order to evaluate the quality of the feature extractors, they were input to a dense neural network (fully-connected) to classify each image into two classes: tattoo or non-tattoo. Both datasets created (TattDetectB and TattDetectF) were used together, for training and testing with the four DNNs mentioned before. A 10-fold cross-validation procedure was used, and the average accuracy is reported in Table 8.

As shown, results were quite similar to each other, since all feature extractors achieved good results. Therefore, since the difference between the performances of the feature extractors was irrelevant, Inception-v3 was simply chosen for the next tattoo detection experiments.

Table 8 – Results for feature extractors evaluation.

Feature Extractor	F1-Score
Inception V3	0.965
VGG-16	0.969
VGG-19	0.956
SqueezeNet	0.962

Source: own author.

4.1.2 Evaluation of the Effect of Data Augmentation

In the literature, it is well-known that data augmentation is a valuable strategy for improving the quality of the classifier (AQUINO *et al.*, 2017), especially for improving its generalization capability. Therefore, experiments were done to answer the question: “Does data augmentation applied to the training set improves the detection capability of the classifiers?”

First, an “internal” baseline must be established. For this purpose, two experiments were done: first training with TattDetectB and testing with TattDetectF and, then, vice-versa. Results are shown on the left side of Table 9.

Next, two new groups of datasets were created. In the first, by augmenting each original image 5 times, we created a dataset with 6,000 images (the 1,000 original images plus 5,000 augmented images). In the second, by augmenting 12 times each original image, we created a training dataset with 13,000 images. The same data augmentation procedures were applied to both, tattoo and non-tattoo images, of the original datasets. Therefore, the following new datasets were created: TattDetectB_Aug6, TattDetectF_Aug6, TattDetectB_Aug13, and TattDetectF_Aug13.

For the first case, 5 randomly chosen transformations were applied, out of the 12 following ones: zoom, vertical mirroring, horizontal mirroring, rotation, warp perspective, Poisson random noise, Gaussian random noise, salt and pepper random noise, random contrast and brightness, Gaussian blur and a bilateral filter. For the second case, all the above-mentioned transformations were applied once each.

Then, using the new augmented datasets, we aimed at verifying if data augmentation can improve the baseline results. That is, if the use of an augmented dataset increases the generalization ability of the classifier. Therefore, four experiments were run, and results are shown on the right side of Table 9.

Comparing the accuracy (recall that all classes have a balanced number of samples) of

Table 9 – Results for the data augmentation tests.

Baseline			Augmented		
Training	Test	Accuracy	Training	Test	Accuracy
TattDetectB	TattDetectF	93.95%	TattDetectB_Aug6	TattDetectF	94,90%
			TattDetectB_Aug13	TattDetectF	95,10%
TattDetectF	TattDetectB	88.40%	TattDetectF_Aug6	TattDetectB	89,39%
			TattDetectF_Aug13	TattDetectB	90,24%

Source: own author.

the baseline and the augmented experiments, it is inferred that data augmentation does improve the generalization capability of the classifier. The amount of improvement is small, possibly because that the baseline is very high. Also, augmenting more (*_Aug13) the original dataset led to even better results. Comparatively, the classifier trained with TattDetectB_Aug13 was the best performing, and it will be used in the next experiments.

4.1.3 Comparison with Other Dataset

In order to perform an “external” evaluation of the proposed approach, it would be desirable to compare its performance with other datasets published in the literature. However, the direct comparison with those works is not possible because some works published results by training and testing in the same dataset. In most cases, there is no information about how the dataset was split into training and testing datasets or if cross-validation was used instead. Besides, most of the datasets used in the previously published works are no longer available for downloading. We succeeded to find only one dataset with reasonable parameters for testing, the NTU_Flickr dataset, with two unbalanced classes and a large number of images (Table 3). Therefore, the question to be answered is: “How does the proposed approach perform with an external dataset?”.

The results for this experiment are presented in Table 10. We first trained the network with the proposed datasets and, then, tested with the NTU_Flickr dataset. Next, it was trained with the NTU_Flickr dataset and tested with the proposed datasets. Observe that the NTU_Flickr is unbalanced, with more tattoo images than non-tattoo images (see Table 3). Consequently, accuracy is an inadequate performance measure, and the F1-score is used instead. For a fair comparison with the previous results (Table 9), it is necessary to report the corresponding F1-score: when trained with TattDetectB_Aug13 and tested with TattDetectF, the F1-score was 0.95; and when trained with TattDetectF_Aug13 and tested with TattDetectB, the F1 score was 0.90.

Table 10 – Results for the classification using an external dataset.

Training	Test	F1-score
TattDetectB_Aug13	NTU_Flickr	0.78
TattDetectF_Aug13	NTU_Flickr	0.79
NTU_Flickr	TattDetectB	0.53
NTU_Flickr	TattDetectF	0.56

Source: own author.

Based on the results of Table 9, it was used both the classifiers trained on the augmented datasets, i.e., TattDetectB_Aug13 and TattDetectF_Aug13, to classify the NTU_Flickr dataset. Results are shown in Table 10.

When training with the proposed datasets and testing with the NTU_Flickr, results were reasonably good, despite the results were less than 10% lower than those achieved in Table 9. On the other hand, when training with NTU_Flickr and testing with the proposed datasets, the results showed a large drop in performance.

To investigate the possible reasons for these differences in performance, Figure 32 shows the confusion matrices for these experiments. In (a), results are seemingly relatively balanced, although in (b) they are not, with many tattoo images being classified as non-tattoo. A visual inspection of the TattDetectF_Aug13 dataset, regarding tattoo images classified as non-tattoos, indicated that they were either tiny tattoos or tattoos covering a large part of the body. This fact suggests that the TattDetectB_Aug13 dataset has a wider range of tattoo sizes in the images, compared with the TattDetectF_Aug13.

Figure 32 – Confusion matrices for the experiments with an external dataset.

a) TattDetectB_Aug13 - NTU_Flickr				b) TattDetectF_Aug13 - NTU_Flickr					
		Predicted				Predicted			
		non_tattoo	tattoo	total			total		
Actual	non_tattoo	77.5%	21.1%	4260	Actual	non_tattoo	68.4%	4.6%	4260
	tattoo	22.5%	78.9%	5740		tattoo	31.6%	95.4%	5740
total		3814	6186	10000	total		5953	4047	10000
c) NTU_Flickr - TattDetectB				d) NTU_Flickr - TattDetectF					
		Predicted				Predicted			
		non_tattoo	tattoo	total			total		
Actual	non_tattoo	95.6%	44.2%	1000	Actual	non_tattoo	98.8%	43.2%	1000
	tattoo	4.4%	55.8%	998		tattoo	1.2%	56.8%	1000
total		228	1770	1998	total		246	1754	2000

Source: own author.

Observing the results of the confusion matrix in (c) and (d), a systematic unbalance

was found. The network trained with NTU_Flickr dataset classified wrongly as tattoo 44.2% of the non-tattoos images of the TattDetectB dataset, and 43.2% of the TattDetectF dataset. The reasoning for this requires a visual inspection of the NTU_Flickr dataset. It was built using only human images in the tattoo class, and random images in the non-tattoo class, i.e., images of animals, sights, objects, drawings, cars, flowers, etc. On the other hand, the proposed datasets were built using only images of people and human body parts, with and without tattoos. This fact misled the classifier trained with the NTU_Flickr to classify anything human-like as the tattoo class.

4.1.4 Qualitative Analysis

Finally, a qualitative analysis was carried out with the objective of verifying in which scenarios the proposed approach had classification errors, and answer the question “Is it possible to identify a pattern in wrongly identified tattoos and non-tattoos?”. Such analysis could shed a light into which sort of tattoos and non-tattoos are more difficult to be classified.

The confusion matrices from two of the best-performing train-test pairs of Table 9 are shown in Figure 33. On the one hand, when trained with TattDetectB_Aug13 and tested with TattDetectF more non-tattoos were wrongly classified as tattoos (8.47%) than the opposite (0.66%). On the other hand, when trained with TattDetectF_Aug13 and tested with TattDetectB more tattoos were wrongly classified as non-tattoos (16.19%) than the opposite (0.25%).

Although the results are obtained in terms of overall accuracy, this qualitative analysis helps to realize that the datasets still deserve a little more attention, especially regarding their image diversity, since this is, possibly, the cause of the asymmetry in the results reported above.

In the case of the model trained with the TattDetectB_Aug13 dataset, which had a larger error when classifying non-tattoos, the dataset was inspected. We observed that the non-tattoo part of the dataset is composed of many clean images, with a large number of images with light backgrounds, and without much visual pollution. On the other hand, the wrongly classified non-tattoo images had more colorful backgrounds or people wearing more colorful clothes with details.

The same qualitative analysis was done with the TattDetectF_Aug13 dataset. We found that the wrongly classified images were those that had tattoos with fewer details, smaller in size to the image or less colorful, a class of images with fewer samples in the trained dataset.

To illustrate the qualitative nature of this analysis, Fig. 34 brings some samples of images

Figure 33 – Confusion matrices for different training and testing datasets.

a) TattDetectB_Aug13 - TattDetectF					b) TattDetectF_Aug13 - TattDetectB				
		Predicted					Predicted		
		non_tattoo	tattoo	total			non_tattoo	tattoo	total
Actual	non_tattoo	99.34%	8.47%	1000	Actual	non_tattoo	83.81%	0.25%	1000
	tattoo	0.66%	91.53%	1000		tattoo	16.19%	99.75%	1000
	total	914	1086	2000		total	1192	808	2000

Source: own author.

that were incorrectly classified as tattoos. It was possible to notice the following: elements with high contrast (a, b), colorful backgrounds (c, d, e, h), many details (h, i), people wearing colorful clothes (j, k), confused or blurred images (f), images with some colored element different from the rest of the image (g – k).

Figure 34 – Examples of non-tattoos wrongly classified as tattoos.



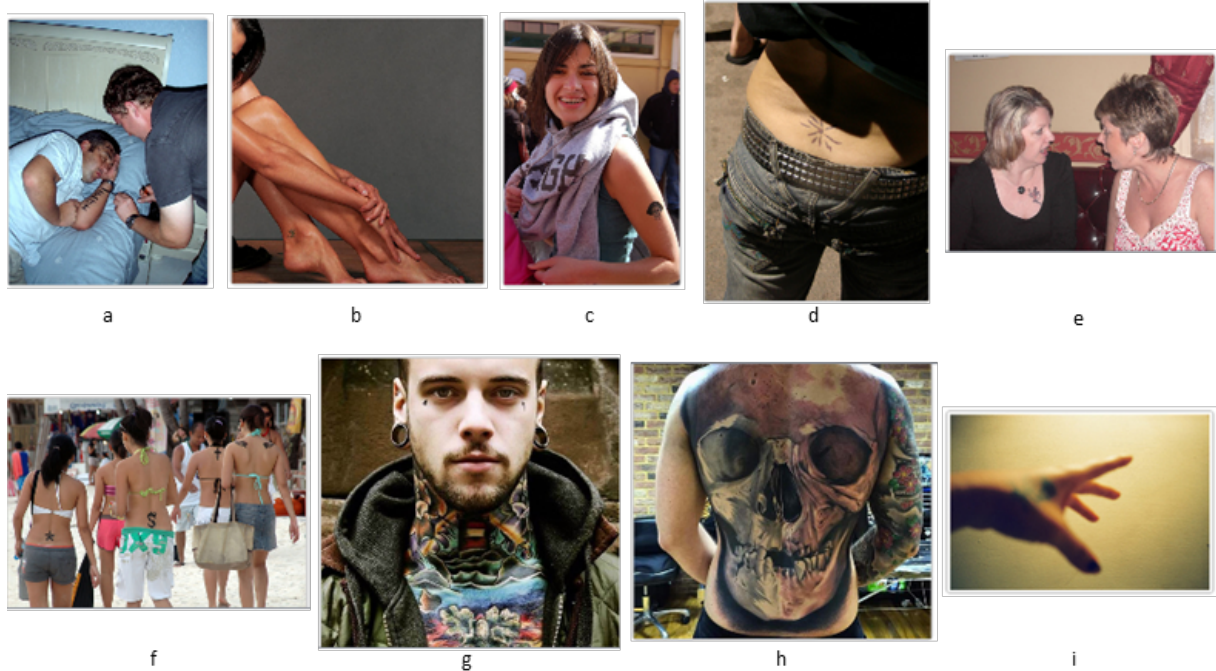
Source: own author.

Similarly, Fig. 35 brings samples of images that had tattoos but were classified as non-tattoos. In general, errors were due to the small size of the tattoo, especially when the tattoo image size were hidden or tiny (a – f), with many people (e, f), person full of tattoos (g, h) or unfocused (i).

4.2 TATTOO LOCATION

The experiments in tattoo location were carried out using scripts in the Python language, running in a Linux Ubuntu environment with Intel(R) Core(TM) i7-9700K CPU @ 3.60GHz

Figure 35 – Examples of tattoos wrongly classified as non-tattoos.



Source: own author.

processor, 64 GB of memory and NVIDIA TITAN Xp GPU with 12Gb of memory.

The process was based in two main experiments:

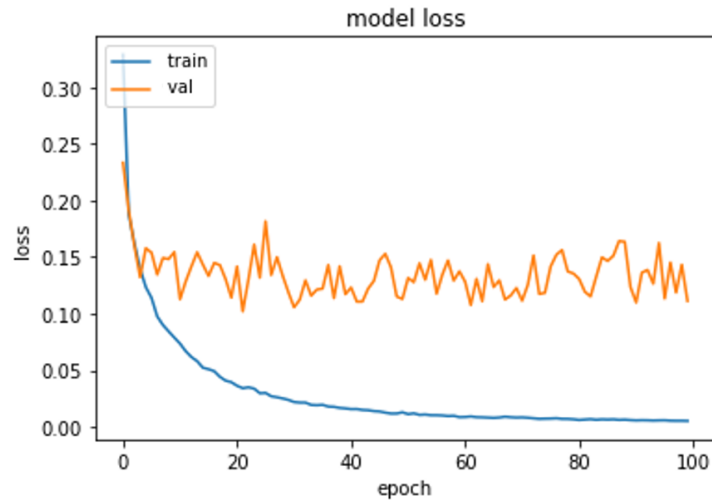
1. Evaluation of the Mask R-CNN with an initial dataset to verify the network's adherence to the tattoo location problem.
2. Application of the Mask R-CNN to a large dataset to evaluate the convergence capability and a fine tune the model to improve results.

4.2.1 Evaluation of the Mask R-CNN for Tattoo Location

The first experiment aimed to answer the question: "Is the Mask R-CNN network suitable for tattoo location?". In other words, this experiment was performed to evaluate if, even using a small dataset, Mask R-CNN can achieve promising results for the tattoo location problem.

For this evaluation, we applied the model presented in Section 3.2.1 to the dataset TattLocA (Table 6), divided into 1,350 images for training, and 333 images for validation. The model was trained with 100 epochs and keeping the hyper-parameters fixed with the default values (Appendix B). As result, Figure 36 shows the evolution of the loss during training.

Figure 36 – Loss to tattoo dataset with default Mask R-CNN parameters.



Source: own author.

The objective of this first experiment was to verify if the Mask R-CNN network would reach an acceptable initial convergence, giving some perspective of its robustness and adaptability to the tattoo data.

Observing Figure 36, although the loss curve for the validation dataset shows an average stationarity and did not converge with the training curve, its average value of 13.44% can be considered a good starting point. Therefore, even though the loss curve is not showing a real convergence, its initial level did not lead to an overfitting nor was it extremely high, which was expected to classify it for the next experiments.

Finally, with this initial result it is possible to continue with the next experiments, aiming at achieving even better results for tattoo location. In other words, it is possible to consider that this initial experiment showed that Mask R-CNN has a great potential to be used in the tattoo location problem.

4.2.2 Evaluation of Mask R-CNN and Fine-Tuning

Given that the initial experiment showed promising results, the following experiment aimed to answer the question: “Does increasing the training dataset and fine-tuning the hyper-parameters of the model lead to improve the results in terms of accuracy and generalization to the tattoo location problem?”

In this group of experiments, the TattLocB dataset (Table 7) was used. This dataset is composed by 4,071 images organized in terms of tattoo size, fill color and style.

The main difficulty in using Mask R-CNN network for tattoo location was the identification and adjustment of its more than 40 hyper-parameters used both in location and segmentation tasks, and determine which of them have more positive impact in training the model. Appendix B shows Mask R-CNN' hyper-parameters, including, for instance learning rate, learning momentum and weight decay, and parameters related to train Region Of Interest (ROI) per image, ROI positive ratio and vectors of RPN anchor scales, backbone strides, among others. Since it is not possible to previously know which set of parameters is dependent or independent, a series of tests was performed until obtain a set of satisfactory hyper-parameters as presented in Appendix A.

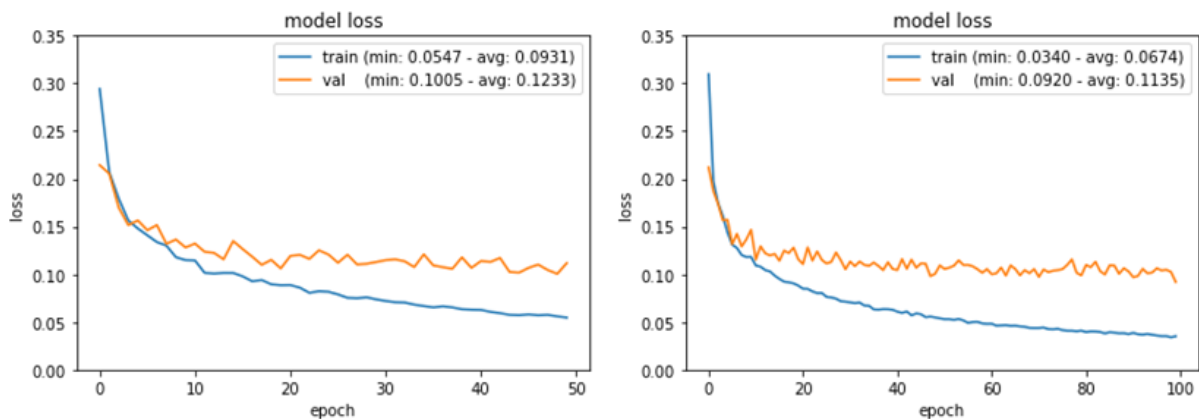
After these experiments, the best set of hyper-parameters found are shown in Table 11. Using such parameters, we obtained a loss average of 0.1135, as shown in Figure 37. Also, the model reached mAP of 0.893 with dvAP of 0.266 in the training dataset, and mAP of 0.761 with dvAP of 0.223 in the validation dataset (Table 12).

Table 11 – Hyper-parameters tested for Mask R-CNN tattoo location.

Hyper-Parameter	Value
Epochs	50
Validation Steps	407
Backbone Strides	4, 8, 16, 32, 64, 96, 128
RPN Anchor Scale	32, 64, 128, 256, 512, 768, 1024
Weight Decay	0.5
Train BN	True
ROI Positive Ration	0.7
Train ROIs per Image	512
Detection NMS Threshold	0.7
Image Min Dim	128
Max GT Instances	120
Detection Max Intances	120

Source: own author.

Figure 37 – Loss to tattoo dataset with fine tuning Mask R-CNN.



Source: own author.

Table 12 – Final results for Mask R-CNN tattoo location.

Evaluation	Train dataset	Validation dataset
Minimum loss	0.034	0.092
Average loss	0.067	0.113
mAP	0.893	0.761
dvAP	0.266	0.223

Source: own author.

Because of the number of hyper-parameters and the high computational cost to train the model, an automated factorial experiment was not possible to be done. However, the results obtained showed that Mask R-CNN is quite adequate for the tattoo location problem.

Although these results cannot be directly compared with the previously results presented in literature (see Table 4) because they used different datasets and quality metrics, the results here presented achieved are very promising, considering both the model and the dataset.

In other words, the process of increasing the datasets, diversifying images in terms of sizes and styles, and performing an adequate fine-tuning in the model parameters, brought more comprehensive results with better assertiveness and generalization for the tattoo location problem as a whole.

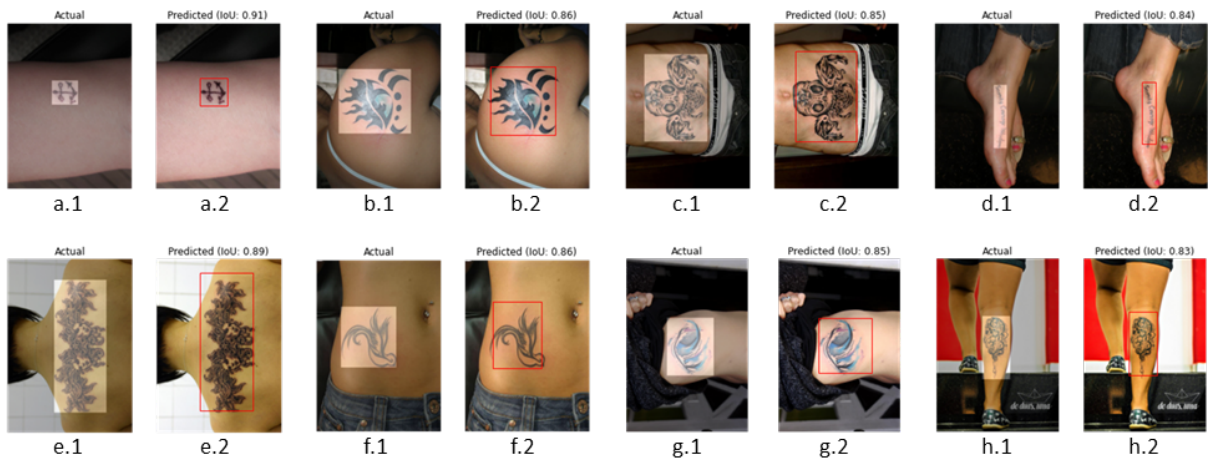
4.2.3 Qualitative Analysis

Finally, a qualitative analysis was carried out with the objective to find more evidence about the characteristics of the images that were easily located, and those that took more difficulty for being located. Therefore, it is aimed at answering the question: “Is it possible to identify characteristics in the tattoos that were best and worse located?”.

Figure 38 presents a collection of images that obtained high IoU values using the model’s validation dataset. In these images, it was possible to observe that the tattoos had a prominent position in the image, and were clearly located within the person’s skin region. Also, the tattoo images were not too large, regarding the total area of the image, and did not have a lot of pollution around them. These characteristics may have favored better results for tattoo location and point to a well-behaved image model for the tattoo location scenario. Regarding the tattoo style, coloring and size, none of them presented a disadvantage in this scenario, as long as they were well positioned and with less noise around them.

On the other hand, Figure 39 presents some examples of tattoos that obtained low IoU values for the bounding box generated by the trained model. These images, in general, consisted

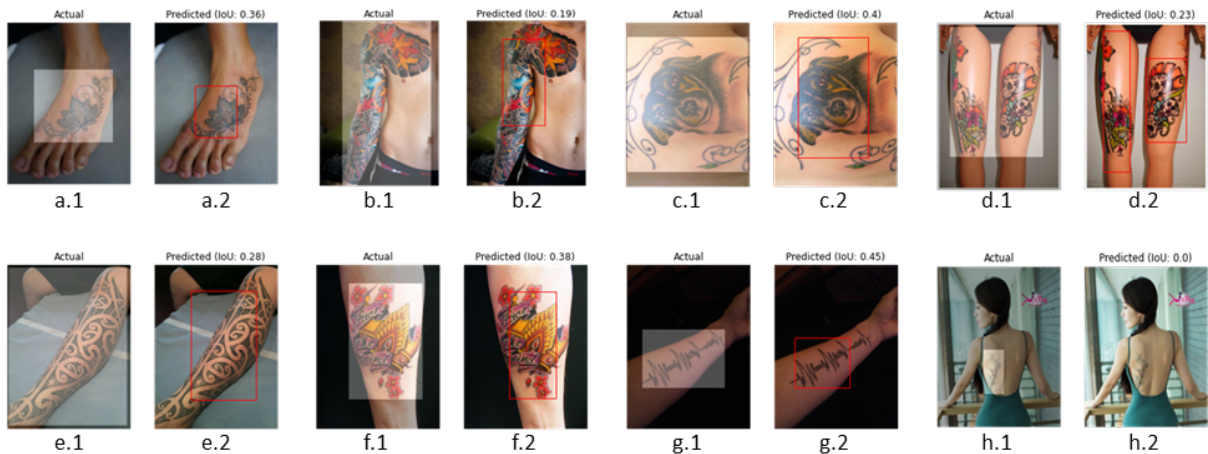
Figure 38 – Sample of good location bounding boxes.



Source: own author.

of tattoos that occupied a large region of the image (see “b.1”, for instance), spread across the contour region of the skin with the background of the image (“a.1”, “b.1”, “e.1”, “f.1”), were in a diagonal position in relation to the image orientation (“a.1”, “e.1”, “g.1”), had poor lighting (“g.1”), had less prominent information around the main image, as in example “c.2”, in which the center of the tattoo is well-defined, but there is a continuation of detail around the entire image.

Figure 39 – Sample of bad location bounding boxes.



Source: own author.

Also, image “d.1” in Figure 39 presents an example of wrong annotation in the original dataset: there were two tattoos, but the annotation was made with only one bounding box around both. In this case, only one of the tattoos was found, and the IoU calculation was impaired. The tattoo exemplified in “h.1”, presents an example where the tattoo was not located. Probably, this is because it is an example where the colors of the tattoo were confused with the colors of the

clothing and, in addition, it is a quite small tattoo, and the network was unable to locate it despite the correct annotation.

Although the model did not perform well for images with specific features, it performed well in the general scope, and these unsuccessful examples can help to infer which characteristics can be considered for further refinements of the model.

4.3 TATTOO CLASSIFICATION

The experiments in tattoo classification were carried out using scripts in the Python language, running in a Linux Ubuntu environment with Intel(R) Core(TM) i7-9700K CPU @ 3.60GHz processor, 64 GB of memory and NVIDIA TITAN Xp GPU with 12Gb of memory.

Based on the model presented in Section 3.3.1, the main objectives in this experiment were:

1. Establish a baseline by performing simple training on the proposed model.
2. Evaluate the effect of data augmentation on the results.
3. Perform a qualitative analysis on the results.

4.3.1 Baseline Experiments

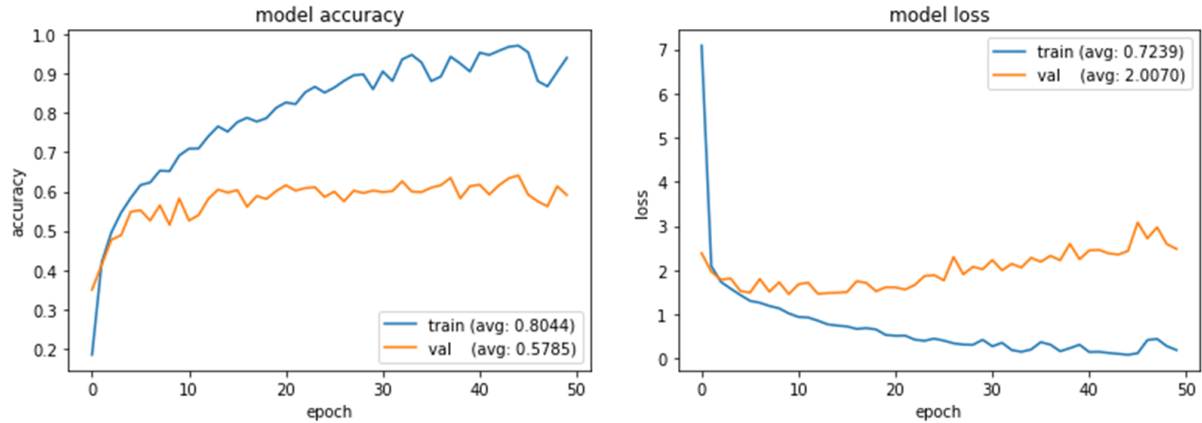
The first experiment in tattoo classification aimed to establish a baseline for the proposed model, and answer the question: “Can the initial proposed model achieve levels of assertiveness relevant to the proposed environment?”.

For this purpose, two experiments were carried out, both with the dataset manually divided in 70% for training and 30% for testing. The initial classifier used had two hidden dense layers with 512 and 256 neurons, respectively. The remaining parameters were: Adam optimizer with learning rate of 0.001, categorical cross-entropy loss function, batch size of 32, using 50 epochs to train.

Firstly, the training returned overfitting. Even doing some changes in the network, such as reducing the number of dense layers, number of neurons in each layer, using dropout, changing optimizer, learning rate and regularization rate, the model still overfitted to the training data. For these experiments, all accuracy and loss curves were similar to represented in Figure 40, and the

respective confusion matrix is presented in Appendix C, Figure 48. The best result was for class “eye” (0.88), and the worst for class “dragon” (0.3).

Figure 40 – Accuracy and loss curves to tattoo classification initial model.



Source: own author.

Overfitting may have occurred because the features created by the model during training and used in the test dataset had low generalization capability, and/or because the datasets did not contain enough images to generalize and separate adequately the classes.

Therefore, and based on the results of the first experiment, a second experiment was performed using a 4-fold cross-validation procedure to get similar proportion of images in training and test datasets, and performing some fine-tuning in the network parameters. After testing different network configurations on the original model (Section 3.3.1) in terms of number of hidden dense layers, number of neurons in each layer, dropout, changing optimizer, learning rates and regularization rates, the best network was composed of two hidden layers, with 1,024 and 512 neurons respectively with L2 regularization rate of 0.01 and ReLU activation function (Figure 41), categorical cross-entropy loss function and Adam optimizer, learning rate of 0.0005 and 100 epochs of training.

Figure 41 – Final tattoo classification dense network.

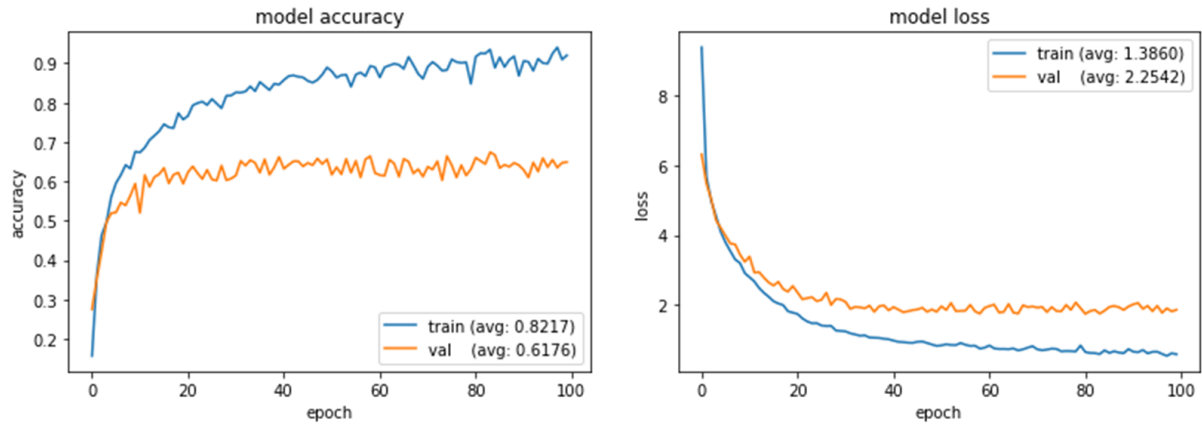
Model: "sequential_35"

Layer (type)	Output Shape	Param #
dense_70 (Dense)	(None, 1024)	1025024
dense_71 (Dense)	(None, 512)	524800
dense_72 (Dense)	(None, 40)	20520
Total params: 1,570,344		
Trainable params: 1,570,344		
Non-trainable params: 0		

Source: own author.

In this experiment, the network achieved a better convergence, compared with the baseline, reaching an average accuracy of 61.68% in the test dataset, with a standard deviation of 1.68%. Figure 42 shows the results achieved by the best fold in the cross-validation. Also, Figure 49 in Appendix C presents its respective confusion matrix. The best result was for classes “star”, “eye”, “down” and “scorpion” (1.00), and the worst for class “leprechaun” (0.26).

Figure 42 – Accuracy and loss curves to tattoo classification model with cross validation.



Source: own author.

Therefore, although the results in this experiment still do not show a very high accuracy, it was possible to establish a new baseline for the next experiments since the model presented a good convergence in accuracy and loss curves.

4.3.2 Evaluation of the Effect of Data Augmentation

Given the results obtained so far, the next question to be answered was: “Can the application of data augmentation bring a significant improvement to the classifier results?”

For this, a new dataset was created by augmenting 4 times each original images, generating a training dataset with 16,000 images. The following transformations were chosen at random and applied to each image: zoom, vertical mirroring, horizontal mirroring, rotation, warp perspective, Poisson random noise, Gaussian random noise, salt and pepper random noise, random contrast and brightness, Gaussian blur and a bilateral filter.

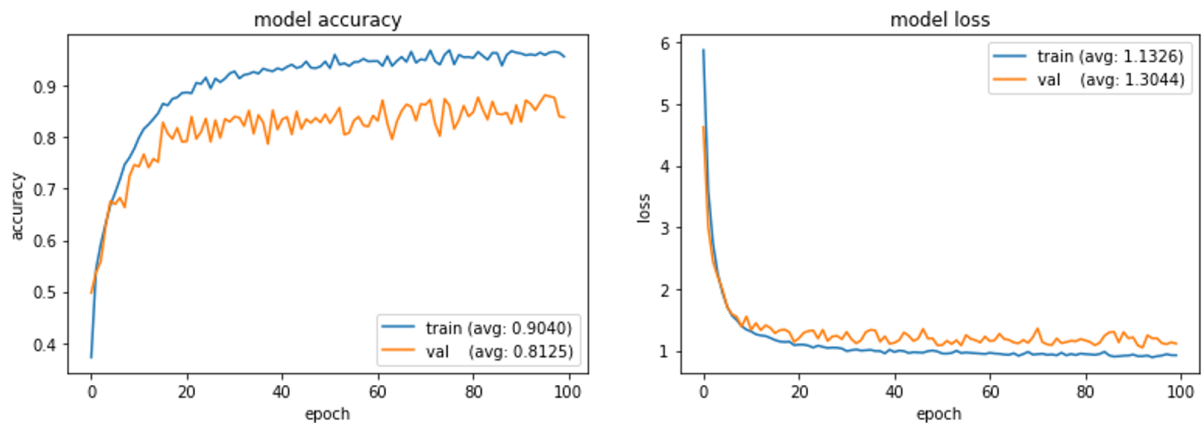
After, using the new augmented dataset, the objective was to verify if the new augmented dataset could improve upon the baseline results. In this experiment, considering the availability of a larger number of images than before, a 10-fold cross-validation procedure was used.

At this point, care was taken so that the images chosen for the test dataset did not have

their respective images enlarged in the training dataset.

For this experiment, the same network built before (Figure 41), with the best performance in the initial experiments, was used. As a result, it was possible to achieve a better convergence, reaching an average accuracy of 85.24%, with a standard deviation of 1.46% on validation dataset. Figure 43 shows the results achieved by the best fold in the cross-validation process. Also, Figure 50 in Appendix C presents its respective confusion matrix. The best result was for classes “car”, “crucifix”, “star”, “eye”, “spider web” and “zombie” (1.00), and the worst for class “wizard” (0.63).

Figure 43 – Accuracy and loss curves to tattoo classification model with cross validation and data augmentation.



Source: own author.

At this point, it is possible to observe that, although the model did not achieved a high accuracy, there was a good convergence of the model as a whole, showing that the data augmentation also had a significant contribution to improve the overall classification results.

In addition, the results indicate that, possibly, if the model is trained with a more robust and diversified database, better results can still be achieved. In fact, and in comparison with the other results presented so far in the literature (Table 5), the results achieved in terms of accuracy, number of classes trained and size of the database, presented relevant results for this research area.

4.3.3 Qualitative Analysis

After establishing a baseline and achieved a model with good accuracy, it was then sought to qualitatively verify the strengths and weaknesses of the developed classifier. Then, the objective was to answer some questions:

- “Is it possible to identify characteristics in the images that make them more difficult to be classified?”
- “How does the model behave with images of other classes not belonging to the training set?”
- “How does the model behave with images having tattoos with more than one element?”
- “How does the model behave with images without tattoos?”

Firstly, observing the confusion matrix 50 in Appendix C, it is possible to identify the classes in which the model had the highest difficulty to classify. Table 13 presents the 5 classes with their respective accuracy in the test dataset.

Table 13 – Classes with the worst classification accuracy.

Class	Accuracy
wizard	0.63
bird	0.71
joker	0.71
women face	0.72
fish	0.74

Source: own author.

The wizard class presented the lowest accuracy (0.63), and false positives were related especially to the clown (0.07), tiger (0.07), gnome (0.05), women face (0.05) and lion (0.05) classes. This example is interesting because it is common to expect that a class that is semantically a person will be confused with other classes that also represent people. However, for this case, classifications such as lion and tiger appeared.

However, looking at some examples of the original images (Figure 44), it is possible to notice that the wizard (“a”, “b”, “c”, “d”, “e” and “f”) has a lot of beard, hat, and other information around the face, which may have contributed to confusing with lion (“g”, “h”, “i”) and tiger (“j”, “k”, “l”) classes.

Another example, however more expected, was the bird class, which had an accuracy of 0.71, and had an error of 0.08 when classified as an angel. On the other hand, the angel class had an accuracy of 0.79 and its largest error was with the bird class (0.06).

The jocker class, in turn, presented a high error rate also with semantically similar classes, being confused with leprechaun (0.07) and zombie (0.07). Similarly, the tiger class presented an accuracy of 0.85 and was more confused with lion (0.06).

Figure 44 – Sample tattoo images of wizard, lion and tiger.



Source: own author.

Therefore, we observed that the classes with the lowest classification accuracy have similar visual characteristics to other classes, which were, possibly, the causes for their high classification error rates. At this point, it should be noted that a precise analysis cannot be performed, since the classification is performed based on the set of features generated via transfer learning with the SqueezeNet network.

On the other hand, the most semantically different classes from the others, such as car, crucifix, star, eye, and spider web, had achieved the best accuracies, what was already expected, given that, visually speaking, they are quite different from the others and not have other classes similar to them.

The next experiment aimed to verify how the model would behave when presented to new images that did not belong to any known classes of the training set. This experiment could give hints on how the classifier would classify the images considering their closest semantic counterparts.

A new test database was created for this experiment, with 10 images for each of the following 16 classes: baby, crown, dagger, dolphin, elephant, falcon, frog, gorilla, mandala, moon, panther, shark, sun, unicorn, witch and wolf. An example of each of them can be seen in Figure 45.

Observe that some of those tattoos have similarity with one or more known classes,

Figure 45 – Example of tattoo classes not trained in the proposed model.



Source: own author.

such as baby, gorilla and witch that are similar to classes related to people. Dolphin and shark classes are similar to fish. Falcon, panther and wolf also have their corresponding classes. The other classes do not have any semantic correspondence with the other trained ones.

Table 14 presents a sample of the results predicted by unknown classes, and a complete experiment results can be seen in Table 18, in Appendix D.

Table 14 – Sample of results of unknown classes classification.

Original	Predicted	Original	Predicted	Original	Predicted	Original	Predicted
baby_01	women face	wolf_01	demon	frog_01	butterfly	sun_01	flower
baby_02	women face	wolf_02	dog	frog_02	heart	sun_02	spider
baby_03	man face	wolf_03	cat	frog_03	leprechaun	sun_03	spider
baby_04	man face	wolf_04	cat	frog_04	leprechaun	sun_04	heart
baby_05	women face	wolf_05	car	frog_05	butterfly	sun_05	angel
baby_06	heart	falcon_06	eagle	mandala_06	taz	moon_06	heart
baby_07	man face	falcon_07	eagle	mandala_07	fish	moon_07	butterfly
baby_08	Jesus	falcon_08	eagle	mandala_08	demon	moon_08	heart
baby_09	women face	falcon_09	eagle	mandala_09	spider	moon_09	heart
baby_10	women face	falcon_10	wings	mandala_10	heart	moon_10	heart

Source: own author.

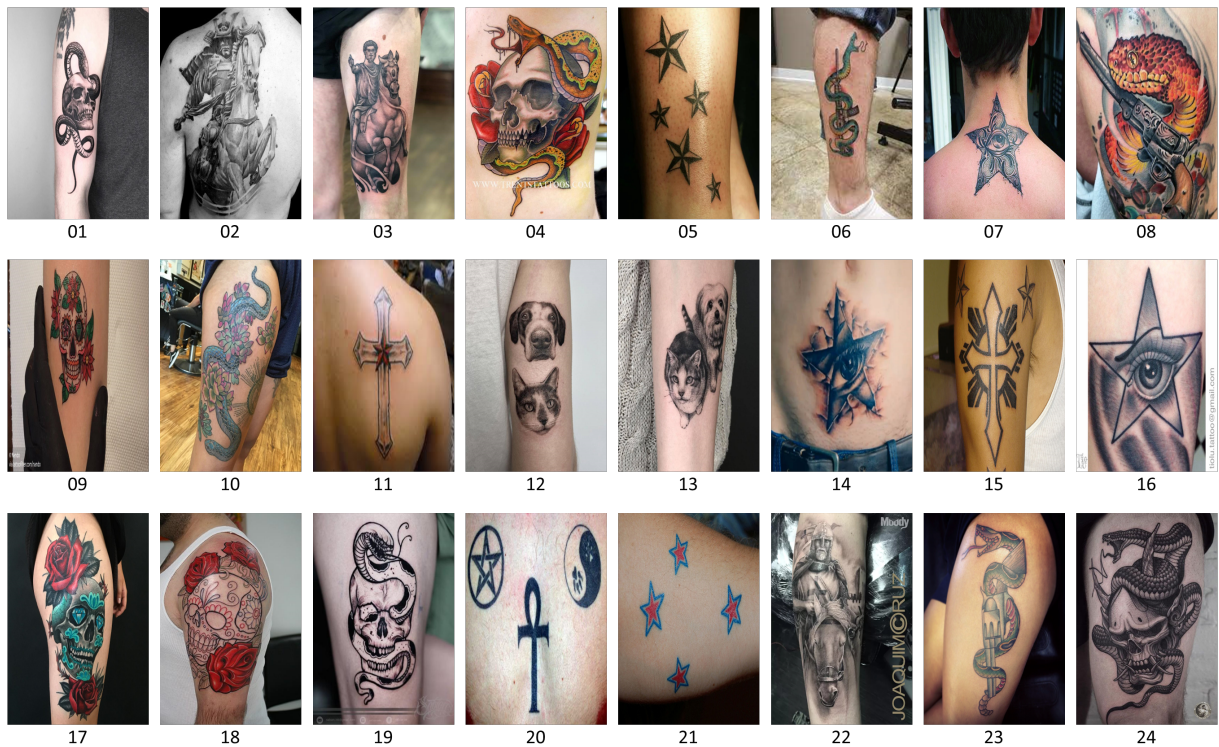
On the one hand, the first two columns present results related to classes semantically similar to the known ones. It is possible to verify that the model returned classes that really have the same semantic meaning. On the other hand, the last two columns present classes that do not correspond to the known classes and, as expected, the results were the most diverse, since the proposed model was not generated to identify items that do not belong to any class in the model. In this case, probably, the predicted class was based on visual similarity, such as the correspondence of sun and spider, or mandala and heart. In this case, each pair would need a visual inspection to identify possible visual similarities.

This experiment allowed a deeper evaluation of the scope and robustness of the trained model, and results showed that the network corresponded well to what was expected, and it was able to predict semantically unknown classes with known classes. As for classes without semantic correlation, it is not possible to predict the classification result.

The last experiment was performed with a set of images with more than one tattoo or with tattoos that have more than one class known by the trained network. The objective was to verify how the classification would behave in a less controlled environment than the one used so far.

Figure 46 presents the images submitted to the classifier, and Table 15 presents the reference of each tattoo, its known classes, the class predicted by the model and a check if the predicted class corresponds to any of the true classes.

Figure 46 – Example of tattoo with more than one class or tattoo.



Source: own author.

We verify that the model does not behave very well in this experiment, since it was able to correctly predict one of the tattoo classes in only 8 out of the 24 images. In the classes in which there was a hit, we noticed that the model prioritized the dominant part of the image. In the other images, probably the overlap between the elements may have tended the model to interpret the image as a whole, leading to an error.

At this point, it is important to observe that the distance between the elements of the

Table 15 – Results for classification in multi classes tattoos.

Reference	Original Classes	Predicted	Is Correct?
01	skull, snake	skull	✓
02	horse, man face	horse	✓
03	horse, man face	angel	✗
04	flower, skull, snake	clown	✗
05	stars	star	✓
06	snake, weapon	women face	✗
07	eye, star	star	✓
08	snake, weapon	fish	✗
09	flower, skull, star	taz	✗
10	flower, snake	dragon	✗
11	crucifix, star	knife	✗
12	cat, dog	dog	✓
13	cat, dog	dog	✓
14	eye, star	bird	✗
15	crucifix, star	joker	✗
16	eye, star	star	✓
17	flower, skull	clown	✗
18	flower, skull	taz	✗
19	skull, snake	skull	✓
20	crucifix, star, yin-yang	heart	✗
21	stars	butterfly	✗
22	horse, man face	women face	✗
23	snake, weapon	guitar	✗
24	skull, snake	dragon	✗

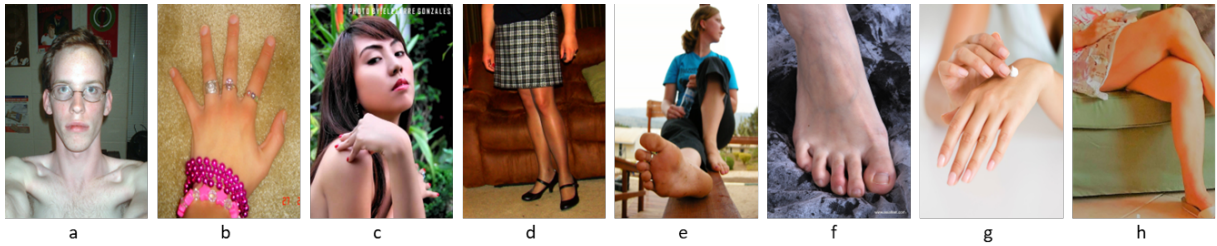
Source: own author.

tattoo (separability) strongly influences the classifier success. For instance, in images 05, 12, 20 and 21, the tattoo location model would be of great help to separate each instance and deliver each one separately to the classifier. However, for the other images, where the elements are intertwined, it would be necessary to perform a semantic segmentation. That is, to submit the localized tattoo to a model capable of identifying and separating the different elements superimposed or grouped in the same scene.

Aiming at as practical application, a final experiment was performed with images without tattoos (Figure 47) submitted to the classifier. As expected, the model returned some class, but that does not make sense from a practical point of view, since the image is known to have no tattoos. Just for reference, the classes returned were respectively: demon (“a”), guitar (“b”), guitar (“c”), spider web (“d”), dragon (“e”), horse (“f”), flower (“g”) and crucifix (“h”).

The objective of this experiment was to highlight the importance of the pre-processing models mentioned before, which aim to increase the accuracy of recognition methods.

Ultimately, this experiment helped to demonstrate that the classifier, as a stand-alone method, may not have confidence for using in real problems under uncontrolled conditions. This assertion reinforces the importance of pre-processing models in the presented tattoo recognition

Figure 47 – Images without tattoos.

Source: own author.

roadmap, which help to detect the presence of a tattoo in the image, locate and separate each tattoo found, reduce the amount of irrelevant information, and segment each object instance. Therefore, such a pre-processing process can deliver noise-free elements to the recognition methods, which include classification and re-identification, in such a way to help to avoid mistakes such as those presented before.

5 CONCLUSIONS AND FUTURE WORKS

The present study aimed to develop computer vision-based methods for problems related to tattoo image processing. More specifically, the problems of detecting, localizing and classifying tattoos in images.

The first two problems approached here, detection and location, are related to preprocessing part of the tattoo recognition roadmap, and the last, classification, is part of the tattoo recognition problems (Figure 19).

From an application point of view and, specifically in applications for public security, new approaches for tackling real-world tattoo recognition can significantly contribute to identify individuals (criminals, or not) and, thus, contribute to enhance public security.

This research project obtained significant results, since it proposed new approaches for the problems addressed, with robust solutions with potential use in the real-world.

The first problem addressed was the tattoo detection, and we used the transfer-learning approach with DNNs and a dense neural network as a classifier, with 10-fold cross-validation. All the tested models had good performance and similar results, reaching an average accuracy of 96.82%.

For this problem, data augmentation was applied to improve the robustness of the classification process. A small difference in performance (~5%) was observed using TattDetectB_Aug13 and TattDetectF_Aug13 for feature extraction, and TattDetectF and TattDetectB for testing, respectively. Results suggested that the datasets proposed in this work are more realistic than NTU_Flickr, keeping in mind that tattoo identification only makes sense in images of humans. It is possible that this fact contributed to reach better results.

However, it is important to emphasize that a quantitative comparison of tattoo detection performance with other works is not possible due to the methodological differences between works. It should be pointed, also, that the former studies with tattoos were based on the Tatt-C dataset, provided by NIST, which was discontinued over time, and it is no longer available for download. Currently, the lack of standardized datasets for tattoo detection is a great drawback for this area of research.

Consequently, future work will include the expansion of all the datasets proposed in this work, with increased diversity and quality, so that, once put in the public domain, more research in this area will be fostered. Also, a more in-depth study about the effect of image properties,

such as sizes, proportion of the tattoo in the image, illumination, effect of colors, effect of the complexity of the tattoos, etc., will be interesting research directions to be sought in the near future.

The second problem addressed was the tattoo location and, for this, we used the Mask R-CNN network. The mAP values achieved were 0.893 and 0.761 for the training and validation bases, which is quite significant for this problem.

Two datasets were created for this study, TattLocA and TattLocB, such that the latter is much larger than the former. Experiments show the importance of the size and diversity of the dataset used for training the model, since better results were obtained with TattLocB.

Similarly to the tattoo detection problem, the lack of publicly available datasets precluded comparisons with other published results. Also, published works in this area used different quality metrics used in this work.

The main difficulty encountered in using the MaskR-CNN network was the performance of fine-tuning, given the large amount of existing hyper-parameters and, given the high computational cost taken for each test. Therefore, a factorial test was not performed.

As future works, the creation of larger and more diversified datasets could contribute to the development of the model robustness, as well as a study with data augmentation. A more in-depth study of the hyper-parameters of the network would also be helpful to achieve even better results.

The last problem addressed was the tattoo classification, using a transfer approach, based on the SqueezeNet DNNs. A specific dataset for tattoo classification was created, named TattClass40. For this 40-classes classification problem, an average accuracy of 85.24% was achieved.

A qualitative analysis helped to demonstrate that the model is robust and comprehensive enough to also classify tattoos of unknown classes, with some semantic similarity with the classes of the model. However, for images with complex, intertwined tattoos or with multiple tattoos, the model showed to be limited.

Once again, it was not possible to compare the results obtained with the others in the literature. However, in this case, the main factor was the number of classes of the dataset. As far as the current search has gone, no study covering 40 classes of tattoos has been found, since previous studies were limited up to 5 classes. Even so, the accuracy achieved here is superior to that presented by previous smaller studies (Table 5).

Furthermore, the qualitative experiments performed demonstrated the importance of the pre-processing methods, including detection, localization and, especially, segmentation, to provide more assertive results in this research area.

Therefore, future works will address the segmentation problem of tattoo images. Also, the development of databases is a relevant topic, both in increasing the number of images of the current dataset, and increasing the number of classes. The open world classification approach is also an important topic to be developed, allowing to create models not only capable of identifying the trained classes, but also recognizing objects that are not part of the set of known classes.

In addition to the developed models, other relevant contributions can be added. Initially, the study brings a new set of high-quality tattoo datasets for tattoo detection, tattoo location and tattoo classification. These datasets are highly complex to obtain, classify, annotate and organize, which is not a trivial task in this context. As previously mentioned, the main datasets used in the literature are no longer available and, in any case, those presented here were developed based on a methodology that had not been used before, in addition to have a high number of images than others cited in the literature.

The second important point to be studied in the future are quality metrics. In the literature review presented in Chapter 2, there is no consensus on which are the best metrics to be used for each of the several tattoo image processing problems. Thus, this is an area that is still open for research aiming at establishing standards.

Another important contribution of this study is related to the tattoo recognition roadmap (Figure 19). The studies carried out here show the relationship among some roadmap items, including the importance of pre-processing methods for the recognition tasks. In addition, there are still very few studies that correlate the methods with each other, and many of the problems presented there still do not have any specific study published in the scientific literature.

It is emphasized that, not only for the problems approached here, but for all problems related to tattoo image processing, the development of robust datasets is a very relevant issue, since they are very scarce and, when found, they have low quality.

Overall, the development of models for problems related to tattoo recognition is an area that is open for research and has many important challenges. New methods based on computer vision, especially those involving deep learning, have great potential to help the evolution of this research area.

REFERENCES

- ABAZA, Ayman; ROSS, Arun; HEBERT, Christina; HARRISON, Mary Ann F.; NIXON, Mark S. A survey on ear biometrics. **ACM Computing Surveys**, v. 45, n. 2, p. 1–35, 2013.
- ABDELWHAB, Abdelgader; VIRIRI, Serestina. A survey on soft biometrics for human identification. *In: Machine Learning and Biometrics. [S.l.]: InTech, 2018. p. 37–56.*
- ACADEMY, Data Science. **Deep Learning Book. [S.l.]: Data Science Academy, 2022.** <https://www.deeplearningbook.com.br>.
- ACTON, Scott T.; ROSSI, Adam. Matching and retrieval of tattoo images: Active contour CBIR and glocal image features. *In: Proc. IEEE Southwest Symposium on Image Analysis and Interpretation. [S.l.: s.n.], 2008. p. 21–24.*
- ALI, Mouad; GAIKWAD, Ashok; YANNAWAR, Pravin. A review: Palmprint recognition process and techniques. *In: . [S.l.: s.n.], 2018. v. 13, p. 7499–7507.*
- AQUINO, N. M. R.; GUTOSKI, M.; HATTORI, L. T.; LOPES, H. S. The effect of data augmentation on the performance of convolutional neural networks. *In: Proc. of XIII Brazilian Congress on Computational Intelligence. Niterói, RJ: ABRICOM, 2017.*
- BACCHINI, Fabio; LORUSSO, Ludovica. A tattoo is not a face. ethical aspects of tattoo-based biometrics. **Journal of Information, Communication and Ethics in Society**, Emerald, v. 16, n. 2, p. 110–122, 2017.
- BOCHKOVSKIY, Alexey; WANG, Chien-Yao; LIAO, Hong-Yuan Mark. **YOLOv4: Optimal Speed and Accuracy of Object Detection.** arXiv, 2020. Available at: <https://arxiv.org/abs/2004.10934>.
- BORRA, Subba Reddy; REDDY, G. Jagadeeswar; REDDY, E. Sreenivasa. A broad survey on fingerprint recognition systems. *In: Proc. IEEE International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET). [S.l.: s.n.], 2016. p. 1428–1434.*
- BOZINOVSKI, Stevo. Reminder of the first paper on transfer learning in neural networks, 1976. **Informatika**, Slovenian Association Informatika, v. 44, n. 3, Sep. 2020.
- BROWNLEE, Jason. **A Gentle Introduction to Computer Vision.** 2019. Machine Learning Mastery. Available at: <https://machinelearningmastery.com/what-is-computer-vision/>.

CHEN, Jiansheng; MOON, Yiu-Sang; WONG, Ming-Fai; SU, Guangda. Palmprint authentication using a symbolic representation of images. **Image and Vision Computing**, v. 28, n. 3, p. 343–351, 2010.

COVEY, Stephen R. **The 7 habits of highly effective people**. 15. ed. New York, NY": Free Press, 2004.

DANTCHEVA, Antitza; VELARDO, Carmelo; D'ANGELO, Angela; DUGELAY, Jean-Luc. Bag of soft biometrics for person identification. **Multimedia Tools and Applications**, Springer Science and Business Media LLC, v. 51, n. 2, p. 739–777, 2010.

DEMSAR, Janez; CURK, Tomaž; ERJAVEC, Aleš; GORUP, Crt; HOCEVAR, Tomaž; MILUTINOVIC, Mitar; MOZINA, Martin; POLAJNAR, Matija; TOPLAK, Marko; STARIC, Anže; STAJDOHAR, Miha; UMEK, Lan; ZAGAR, Lan; ZBONTAR, Jure; ZITNIK, Marinka; ZUPAN, Blaž. Orange: Data mining toolbox in python. **Journal of Machine Learning Research**, v. 14, p. 2349–2353, 2013.

DENG, Jia; DONG, Wei; SOCHER, Richard; LI, Li-Jia; LI, Kai; FEI-FEI, Li. ImageNet: A large-scale hierarchical image database. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2009. p. 248–255.

DI, X.; PATEL, V. M. Deep tattoo recognition. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Piscataway, NJ, USA: IEEE Press, 2016. p. 119–126.

DI, X.; PATEL, V. M. Deep learning for tattoo recognition. *In: BHANU, B.; KUMAR, A. (Ed.). Deep Learning for Biometrics*. Cham: Springer, 2017, (Advances in Computer Vision and Pattern Recognition). p. 241–256.

DONG, Jinjin; MENG, Xiao; CHEN, Meng; WANG, Zhifang. Template protection based on DNA coding for multimodal biometric recognition. *In: 2017 4th International Conference on Systems and Informatics (ICSAI)*. [S.l.]: IEEE, 2017.

DUBAL, S.S.; BHARADI, V.A. Comparative analysis of various approaches for different biometric traits. *In: Proc. International Conference & Workshop on Electronics & Telecommunication Engineering (ICWET 2016)*. [S.l.]: Institution of Engineering and Technology, 2016. p. 163–168.

EVERINGHAM, Mark; GOOL, Luc Van; WILLIAMS, Christopher K. I.; WINN, John; ZISSERMAN, Andrew. The Pascal Visual Object Classes (VOC) Challenge. **International Journal of Computer Vision**, v. 88, n. 2, p. 303–338, 2010.

FANG, Shelley; COVERDALE, John; NGUYEN, Phuong; GORDON, Mollie. Tattoo recognition in screening for victims of human trafficking. **Journal of Nervous & Mental Disease**, Ovid Technologies (Wolters Kluwer Health), v. 206, n. 10, p. 824–827, 2018.

FARRÚS, Mireia. Voice disguise in automatic speaker recognition. **ACM Computing Surveys**, v. 51, n. 4, p. 1–22, 2018.

GIRSHICK, Ross. Fast r-CNN. *In: 2015 IEEE International Conference on Computer Vision (ICCV)*. [S.l.]: IEEE, 2015.

GIRSHICK, Ross; DONAHUE, Jeff; DARRELL, Trevor; MALIK, Jitendra. Rich feature hierarchies for accurate object detection and semantic segmentation. *In: 2014 IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.]: IEEE, 2014.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. **Deep Learning**. [S.l.]: MIT Press, 2016. <http://www.deeplearningbook.org>.

GOODFELLOW, Ian J.; POUGET-ABADIE, Jean; MIRZA, Mehdi; XU, Bing; WARDEFARLEY, David; OZAIR, Sherjil; COURVILLE, Aaron; BENGIO, Yoshua. **Generative Adversarial Networks**. [S.l.]: arXiv, 2014.

GUI, Qiong; RUIZ-BLONDET, Maria V.; LASZLO, Sarah; JIN, Zhanpeng. A survey on brain biometrics. **ACM Computing Surveys**, v. 51, n. 6, p. 1–38, 2019.

GUO, Guodong; MU, Guowang. A framework for joint estimation of age, gender and ethnicity on a large database. **Image and Vision Computing**, Association for Computing Machinery (ACM), v. 32, n. 10, p. 761–770, 2014.

GUTOSKI, M.; RIBEIRO, M.; HATTORI, L. T.; ROMERO, M.; LAZZARETTI, A. E.; LOPES, H. S. A comparative study of transfer learning approaches for video anomaly detection. **International Journal of Pattern Recognition and Artificial Intelligence**, v. 35, n. 5, p. 2152003, 2021.

HAN, H.; LI, J.; JAIN, A. K.; SHAN, S.; CHEN, X. Tattoo image search at scale: Joint detection and compact representation learning. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 41, n. 10, p. 2333–2348, 2019.

HAO, Xing; ZHANG, Guigang; MA, Shang. Deep learning. **International Journal of Semantic Computing**, World Scientific Pub Co Pte Lt, v. 10, n. 03, p. 417–439, Sep. 2016.

HARBERT, Tam. FBI wants better automated image analysis for tattoos [news]. **IEEE Spectrum**, v. 52, n. 9, p. 13–16, 2015.

HE, Kaiming; GKIOXARI, Georgia; DOLLAR, Piotr; GIRSHICK, Ross. Mask r-CNN. *In: 2017 IEEE International Conference on Computer Vision (ICCV)*. [S.l.]: IEEE, 2017.

HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing; SUN, Jian. **Deep Residual Learning for Image Recognition**. [S.l.]: arXiv, 2015.

HEFLIN, Brian; SCHEIRER, Walter; BOULT, T. E. Detecting and classifying scars, marks, and tattoos found in the wild. *In: Proc. IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*. [S.l.: s.n.], 2012. p. 31–38.

HUSSAIN, Mahbub; BIRD, Jordan J.; FARIA, Diego R. A study on CNN transfer learning for image classification. *In: Advances in Intelligent Systems and Computing*. [S.l.]: Springer International Publishing, 2018. p. 191–202.

IANDOLA, Forrest N.; HAN, Song; MOSKEWICZ, Matthew W.; ASHRAF, Khalid; DALLY, William J.; KEUTZER, Kurt. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. **arXiv preprint**, v. 1602.07360v4, 2016.

IBSEN, Mathias; RATHGEB, Christian; FINK, Thomas; DROZDOWSKI, Pawel; BUSCH, Christoph. Impact of facial tattoos and paintings on face recognition systems. **IET Biometrics**, Institution of Engineering and Technology (IET), 2021.

JADERBERG, Max; SIMONYAN, Karen; ZISSERMAN, Andrew; KAVUKCUOGLU, Koray. **Spatial Transformer Networks**. [S.l.]: arXiv, 2015.

JAIN, A.K.; ROSS, A.; PANKANTI, S. Biometrics: A tool for information security. **IEEE Transactions on Information Forensics and Security**, v. 1, n. 2, p. 125–143, 2006.

JAIN, Anil K.; DASS, Sarat C.; NANDAKUMAR, Karthik. Soft biometric traits for personal recognition systems. *In: Biometric Authentication*. [S.l.]: Springer Berlin Heidelberg, 2004. p. 731–738.

JAIN, Anil K.; JIN, Rong; LEE, Jung-Eun. Tattoo image matching and retrieval. **Computer, IEEE**, v. 45, n. 5, p. 93–96, 2012.

JAIN, Anil K.; LEE, Jung-Eun; JIN, Rong. Tattoo-ID: Automatic tattoo image retrieval for suspect and victim identification. *In: Advances in Multimedia Information Processing*. Heidelberg: Springer, 2007. p. 256–265.

JAIN, Anil K.; LEE, Jung-Eun; JIN, Rong; GREGG, Nicholas. Content-based image retrieval: An application to tattoo images. *In: Proc. 16th IEEE International Conference on Image Processing (ICIP)*. [S.l.: s.n.], 2009. p. 2745–2748.

JIawang, Cao; YUAN, Zhu. Tattoo recognition based on triplet GAN. *In: Proc. 37th Chinese Control Conference (CCC)*. [S.l.]: IEEE, 2018. p. 9595–9597.

KARPATHY, Andrej; FEI-FEI, Li. **Deep Visual-Semantic Alignments for Generating Image Descriptions**. [S.l.]: arXiv, 2014.

KATSANIS, Sara H.; CLAES, Peter; DOERR, Megan; COOK-DEEGAN, Robert; TENENBAUM, Jessica D.; EVANS, Barbara J.; LEE, Myoung Keun; ANDERTON, Joel; WEINBERG, Seth M.; WAGNER, Jennifer K. U.s. adult perspectives on facial images, DNA, and other biometrics. **IEEE Transactions on Technology and Society**, Institute of Electrical and Electronics Engineers (IEEE), v. 3, n. 1, p. 9–15, Mar. 2022.

KAUR, Mandeep; JINDAL, Sonika. Survey on offline signature recognition techniques. **International Journal of Engineering Trends and Technology**, v. 36, n. 6, p. 309–313, 2016.

KIM, Joonsoo; LI, He; YUE, Jiaju; RIBERA, Javier; DELP, Edward J.; HUFFMAN, Landis. Automatic and manual tattoo localization. *In: Proc. IEEE Symposium on Technologies for Homeland Security (HST)*. [S.l.: s.n.], 2016. p. 1–6.

KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. ImageNet classification with deep convolutional neural networks. **Communications of the ACM**, Association for Computing Machinery (ACM), v. 60, n. 6, p. 84–90, May 2017.

KUZNETSOVA, Alina; ROM, Hassan; ALLDRIN, Neil; UIJLINGS, Jasper; KRASIN, Ivan; PONT-TUSET, Jordi; KAMALI, Shahab; POPOV, Stefan; MALLOCI, Matteo; KOLESNIKOV, Alexander; DUERIG, Tom; FERRARI, Vittorio. The open images dataset v4. **International Journal of Computer Vision**, Springer Science and Business Media LLC, v. 128, n. 7, p. 1956–1981, 2020.

LABATI, Ruggero Donida; GENOVESE, Angelo; MUÑOZ, Enrique; PIURI, Vincenzo; SCOTTI, Fabio; SFORZA, Gianluca. Biometric recognition in automated border control. **ACM Computing Surveys**, v. 49, n. 2, p. 1–39, 2016.

LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. **Nature**, Springer Science and Business Media LLC, v. 521, n. 7553, p. 436–444, May 2015.

LEE, Jung-Eun; JIN, Rong; JAIN, Anil; TONG, Wei. Image retrieval in forensics: Tattoo image database application. **IEEE Multimedia**, v. 19, n. 1, p. 40–49, 2012.

LEE, Jung-Eun; JIN, Rong; JAIN, Anil K. Rank-based distance metric learning: An application to image retrieval. *In: 2008 IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2008. p. 1–8.

LI, Rui; YANG, Jun. Improved YOLOv2 object detection model. *In: 2018 6th International Conference on Multimedia Computing and Systems (ICMCS)*. [S.l.]: IEEE, 2018.

LIN, Tsung-Yi; MAIRE, Michael; BELONGIE, Serge; HAYS, James; PERONA, Pietro; RAMANAN, Deva; DOLLÁR, Piotr; ZITNICK, C. Lawrence. Microsoft COCO: Common objects in context. *In: Proc. European Conference on Computer Vision ECCV*. [S.l.]: Springer, 2014. p. 740–755.

LURIG, Moritz D.; DONOUGHE, Seth; SVENSSON, Erik I.; PORTO, Arthur; TSUBOI, Masahito. Computer vision, machine learning, and the promise of phenomics in ecology and evolutionary biology. *Frontiers in Ecology and Evolution*, Frontiers Media SA, v. 9, Apr. 2021.

MARCETIĆ, Darijan; RIBARIĆ, Slobodan; STRUĆ, Vitomir; PAVESIĆ, Nikola. An experimental tattoo de-identification system for privacy protection in still images. *In: Proc. 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. [S.l.]: IEEE, 2014. p. 1288–1293.

MARSICO, Maria de; GALDI, Chiara; NAPPI, Michele; RICCIO, Daniel. FIRME: Face and iris recognition for mobile engagement. *Image and Vision Computing*, v. 32, n. 12, p. 1161–1172, 2014.

MCCABE, R. Michael; NEWTON, Elaine M. **American National Standard for Information Systems—Data Format for the Interchange of Fingerprint Facial, & Other Biometric Information – Part 1**. Gaithersburg (MD), USA, 2007.

MENEGHETTI, Diego. **O que significam as principais tatuagens de presidiário?** 2018. Super Interessante. Available at: <https://super.abril.com.br/mundo-estranho/o-que-significam-as-principais-tatuagens-de-presidiario/>.

MIHAJLOVIC, Ilija. **Everything You Ever Wanted To Know About Computer Vision**. 2019. Towards Data Science. Available at: <https://towardsdatascience.com/everything-you-ever-wanted-to-know-about-computer-vision-heres-a-look-why-it-is-so-awesome-e8a58dfb641e>.

MIKOLAJCZYK, Agnieszka; GROCHOWSKI, Michal. Data augmentation for improving deep learning in image classification problem. *In: 2018 International Interdisciplinary PhD Workshop (IIPhDW)*. [S.l.]: IEEE, 2018.

National Research Council. **Biometric Recognition – Challenges and Opportunities**. Washington DC, USA: National Academies Press, 2010.

NGAN, M.; GROTH, P. Tattoo recognition technology - challenge (Tatt-C): an open tattoo database for developing tattoo recognition research. *In: Proceedings of the IEEE International*

- Conference on Identity, Security and Behavior Analysis (ISBA 2015)**. [*S.l.: s.n.*], 2015. p. 1–6.
- NGAN, M.; QUINN, G.; GROTH, P. **Tattoo Recognition Technology - Challenge (Tatt-C) - Outcomes and Recommendations**. Gaithersburg, MD, USA, 2016.
- NGUYEN, Kien; FOOKES, Clinton; JILLELA, Raghavender; SRIDHARAN, Sridha; ROSS, Arun. Long range iris recognition: A survey. **Pattern Recognition**, v. 72, p. 123–143, 2017.
- NITHYA, Alice; LAKSHMI, C. Iris recognition techniques: A literature survey. **International Journal of Applied Engineering Research**, v. 10, p. 32525–32546, 2015.
- O'MAHONY, Niall; CAMPBELL, Sean; CARVALHO, Anderson; HARAPANAHALLI, Suman; HERNANDEZ, Gustavo Velasco; KRPALKOVA, Lenka; RIORDAN, Daniel; WALSH, Joseph. Deep learning vs. traditional computer vision. *In: Advances in Intelligent Systems and Computing*. [*S.l.*]: Springer International Publishing, 2019. p. 128–144.
- PEREZ, Luis; WANG, Jason. **The Effectiveness of Data Augmentation in Image Classification using Deep Learning**. [*S.l.*]: arXiv, 2017.
- RATTANI, Ajita; DERAKHSHANI, Reza. Ocular biometrics in the visible spectrum: A survey. **Image and Vision Computing**, v. 59, p. 1–16, 2017.
- REDMON, Joseph; DIVVALA, Santosh; GIRSHICK, Ross; FARHADI, Ali. **You Only Look Once: Unified, Real-Time Object Detection**. [*S.l.*]: arXiv, 2015.
- REDMON, Joseph; FARHADI, Ali. **YOLO9000: Better, Faster, Stronger**. [*S.l.*]: arXiv, 2016.
- REDMON, Joseph; FARHADI, Ali. **YOLOv3: An Incremental Improvement**. [*S.l.*]: arXiv, 2018.
- REN, Shaoqing; HE, Kaiming; GIRSHICK, Ross; SUN, Jian. Faster r-CNN: Towards real-time object detection with region proposal networks. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, Institute of Electrical and Electronics Engineers (IEEE), v. 39, n. 6, p. 1137–1149, Jun. 2017.
- ROHITH, Muraleedharan M.; BELCHER, William R.; ROY, Jyotirmoy; ABRAHAM, Shelby O.; CHAKRABORTY, Pooja; NANDANIYA, Nidhi J.; JOHNSON, Abraham. Tattoo in forensic science: An indian perspective. **Journal of Forensic and Legal Medicine**, Elsevier BV, v. 74, p. 102022, 2020.

ROMERO, M.; GUTOSKI, M.; HATTORI, L. T.; RIBEIRO, M.; LOPES, H. S. Soft biometrics classification in videos using transfer learning and bidirectional long short-term memory networks. **Learning and Nonlinear Models**, v. 18, n. 1, p. 47–59, 2020.

ROMERO, M.; GUTOSKI, M.; HATTORI, L. T.; RIBEIRO, M.; LOPES, H. S. A study of the influence of data complexity and similarity on soft biometrics classification performance in a transfer learning scenario. **Learning and Nonlinear Models**, v. 18, n. 2, p. 56–65, 2020.

ROSEBROCK, Adrian. **Intersection over Union (IoU) for object detection**. 2016. Py Image Search. Available at: <https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>.

SCHROFF, Florian; KALENICHENKO, Dmitry; PHILBIN, James. FaceNet: A unified embedding for face recognition and clustering. *In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2015. p. 815–823.

SERDOUK, Yasmine; NEMMOUR, Hassiba; CHIBANI, Youcef. Handwritten signature verification using the quad-tree histogram of templates and a support vector-based artificial immune classification. **Image and Vision Computing**, Elsevier BV, v. 66, p. 26–35, 2017.

SHARMA, Abhilash. Biometric system - a review. **International Journal of Computer Science and Information Technologies**, v. 6, n. 5, p. 4616–4619, 2015.

SHORTEN, Connor; KHOSHGOFTAAR, Taghi M. A survey on image data augmentation for deep learning. **Journal of Big Data**, v. 6, n. 1, p. 1–48, 2019.

SIMONYAN, Karen; ZISSERMAN, Andrew. **Very Deep Convolutional Networks for Large-Scale Image Recognition**. [S.l.]: arXiv, 2014.

SIMONYAN, Karen; ZISSERMAN, Andrew. Very deep convolutional networks for large-scale image recognition. *In: BENGIO, Yoshua; LECUN, Yann (Ed.). Proceedings of 3rd International Conference on Learning Representations*. [S.l.: s.n.], 2015. p. 1–14.

SONI, Urmi; MAHESH, Goyani. A survey on state of the art methods of fingerprint recognition. **Journal of Advanced Research in Dynamical and Control Systems**, v. 4, n. 2, p. 189–200, 2018.

SUN, Z. H.; BAUMES, J.; TUNISON, P.; TUREK, M.; HOOGS, A. Tattoo detection and localization using region-based deep learning. *In: Proceedings of the 23rd International Conference on Pattern Recognition (ICPR)*. Piscataway, NJ, USA: IEEE Press, 2016. p. 3055–3060.

SZEGEDY, Christian; LIU, Wei; JIA, Yangqing; SERMANET, Pierre; REED, Scott; ANGUELOV, Dragomir; ERHAN, Dumitru; VANHOUCKE, Vincent; RABINOVICH, Andrew. Going deeper with convolutions. *In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.]: IEEE, 2015.

SZEGEDY, Christian; VANHOUCKE, Vincent; IOFFE, Sergey; SHLENS, Jonathon; WOJNA, Zbigniew. Rethinking the Inception architecture for computer vision. **arXiv preprint**, v. 1512.00567v3, 2015.

SZELISKI, Richard. **Computer Vision**. [S.l.]: Springer International Publishing, 2022.

TAN, Chuanqi; SUN, Fuchun; KONG, Tao; ZHANG, Wenchang; YANG, Chao; LIU, Chunfang. A survey on deep transfer learning. *In: Artificial Neural Networks and Machine Learning – ICANN 2018*. [S.l.]: Springer International Publishing, 2018. p. 270–279.

UNAR, J.A.; SENG, Woo Chaw; ABBASI, Almas. A review of biometric technology along with trends and prospects. **Pattern Recognition**, Elsevier BV, v. 47, n. 8, p. 2673–2688, 2014.

VERLEKAR, Tanmay Tulsidas; SOARES, Luís Ducla; CORREIA, Paulo Lobato. Gait recognition in the wild using shadow silhouettes. **Image and Vision Computing**, v. 76, p. 1–13, 2018.

VIEJO, Claudia Gonzalez; TORRICO, Damir D.; DUNSHEA, Frank R.; FUENTES, Sigfredo. Emerging technologies based on artificial intelligence to assess the quality and consumer preference of beverages. **Beverages**, MDPI AG, v. 5, n. 4, p. 62, Nov. 2019.

VOULODIMOS, Athanasios; DOULAMIS, Nikolaos; DOULAMIS, Anastasios; PROTOPAPADAKIS, Eftychios. Deep learning for computer vision: A brief review. **Computational Intelligence and Neuroscience**, Hindawi Limited, v. 2018, p. 1–13, 2018.

WEISS, Karl; KHOSHGOFTAAR, Taghi M.; WANG, DingDing. A survey of transfer learning. **Journal of Big Data**, Springer Science and Business Media LLC, v. 3, n. 1, May 2016.

WILBER, Michael J.; RUDD, Ethan; HEFLIN, Brian; LUI, Yui-Man; BOULT, Terrance E. Exemplar codes for facial attributes and tattoo recognition. *In: Proc. IEEE Winter Conference on Applications of Computer Vision*. [S.l.: s.n.], 2014. p. 205–212.

WING, Bradford J. **Data Format for the Interchange of Fingerprint, Facial and Other Biometric Information**. Gaithersburg (MD), USA, 2011.

XU, Q.; GHOSH, S.; XU, X.; HUANG, Y.; KONG, A. W. K. Tattoo detection based on CNN and remarks on the NIST database. *In: Proceedings of the International Conference on Biometrics (ICB)*. Piscataway, NJ, USA: IEEE Press, 2016. p. 1–7.

XU, X.; KONG, A. W. K. A geometric-based tattoo retrieval system. *In: Proceedings of the International Conference on Pattern Recognition (ICPR)*. Piscataway, NJ, USA: IEEE Press, 2016. p. 3019–3024.

ZAHID, Ali Z.Ghazi; AL-KHARSAN, Ibrahim Hasan Mohammed Salih; BAKARMAN, Hesham A.; GHAZI, Muntadher Faisal; SALMAN, Hanan Abbas; HASOON, Feras N. Biometric authentication security system using human DNA. *In: 2019 First International Conference of Intelligent Computing and Engineering (ICOICE)*. [S.l.]: IEEE, 2019.

ZEILER, Matthew D; FERGUS, Rob. **Visualizing and Understanding Convolutional Networks**. [S.l.]: arXiv, 2013.

ZHANG, David; ZUO, Wangmeng; YUE, Feng. A comparative study of palmprint recognition algorithms. *ACM Computing Surveys*, v. 44, n. 1, p. 1–37, 2012.

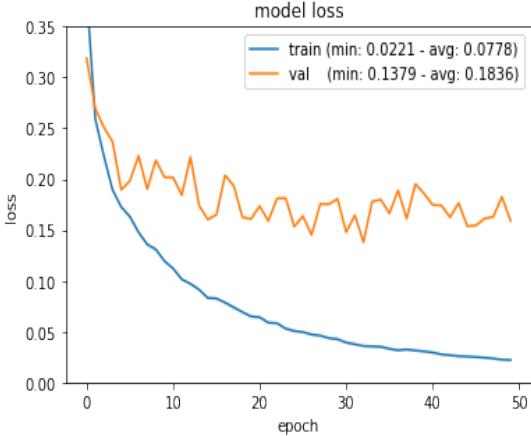
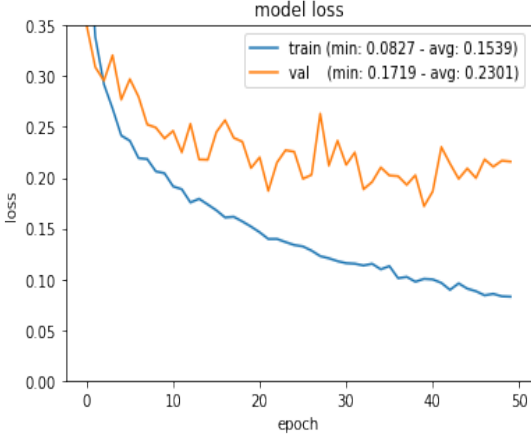
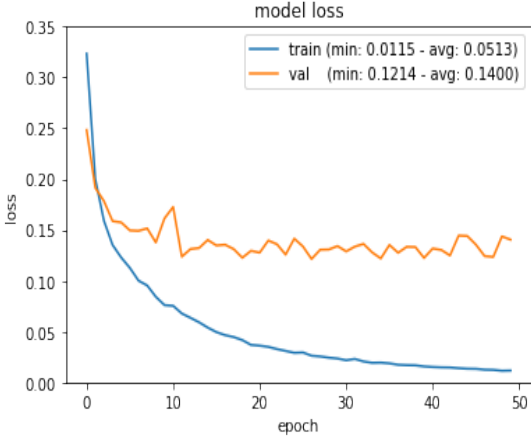
ZHAO, W.; CHELLAPPA, R.; PHILLIPS, P. J.; ROSENFELD, A. Face recognition. *ACM Computing Surveys*, v. 35, n. 4, p. 399–458, 2003.

ZHUANG, Fuzhen; QI, Zhiyuan; DUAN, Keyu; XI, Dongbo; ZHU, Yongchun; ZHU, Hengshu; XIONG, Hui; HE, Qing. **A Comprehensive Survey on Transfer Learning**. [S.l.]: arXiv, 2019.

APPENDIX

APPENDIX A – TATTOO LOCATION - MASK R-CNN FINE TUNING

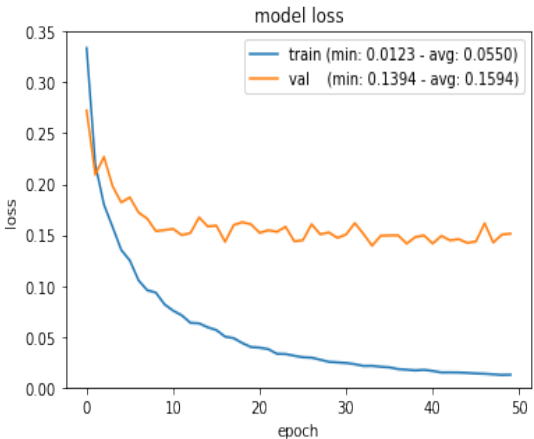
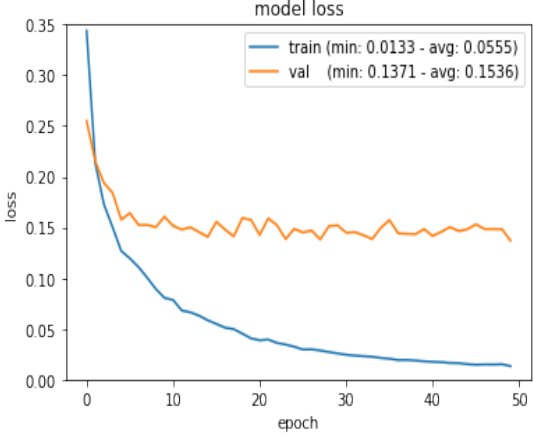
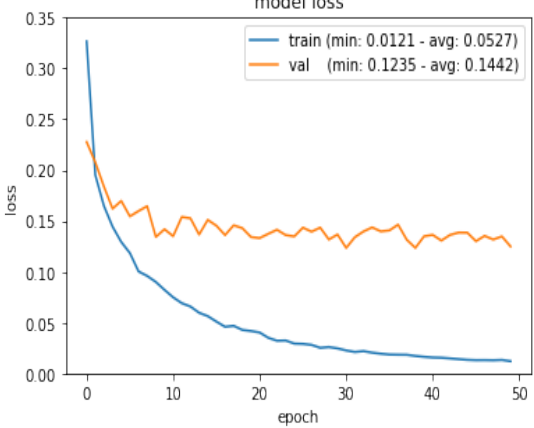
Table 16 – Hyper-parameters tested in Mask R-CNN tattoo location.

Test	Hyper-parameters	Train Loss
01	EPOCHS = 50 LEARNING_MOMENTUM = 0.75 LEARNING_RATE = 0.0005 RPN_NMS_THRESHOLD = 0.6	
02	EPOCHS = 50 LEARNING_MOMENTUM = 0.75 LEARNING_RATE = 0.0001 RPN_NMS_THRESHOLD = 0.6	
03	EPOCHS = 50 LEARNING_MOMENTUM = 0.95 LEARNING_RATE = 0.005 RPN_NMS_THRESHOLD = 0.8	[Generate loss NaN and 0.000e+00 since first epoch.]
04	VALIDATION_STEPS = 200	

(continue)

Table 16 – Hyper-parameters tested in Mask R-CNN tattoo location.

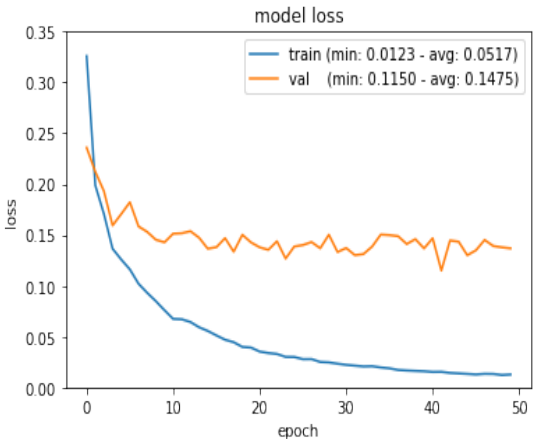
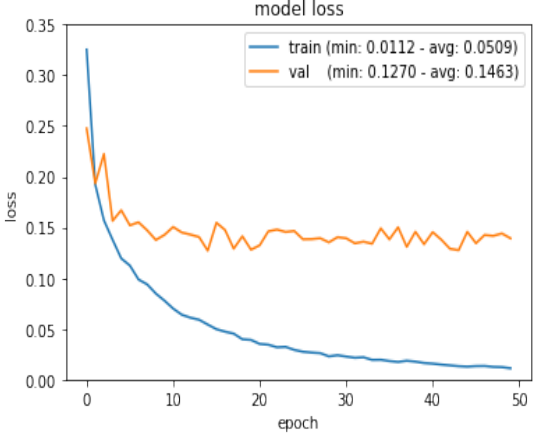
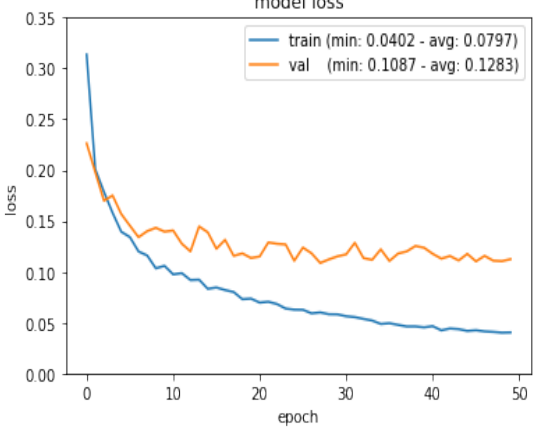
(continuation)

Test	Hyper-parameters	Train Loss
05	EPOCHS = 50 LEARNING_RATE = 0.0005 VALIDATION_STEPS = 333	
06	EPOCHS = 50 LEARNING_RATE = 0.0005 VALIDATION_STEPS = 333 RPN_ANCHOR_SCALES = (64, 128, 256, 512, 768)	
07	EPOCHS = 50 VALIDATION_STEPS = 333 BACKBONE_STRIDES = [4, 8, 16, 32, 64, 96, 128] RPN_ANCHOR_SCALES = (32, 64, 128, 256, 512, 768, 1,024) RPN_ANCHOR_RATIOS = [0.5, 1, 2]	

(continue)

Table 16 – Hyper-parameters tested in Mask R-CNN tattoo location.

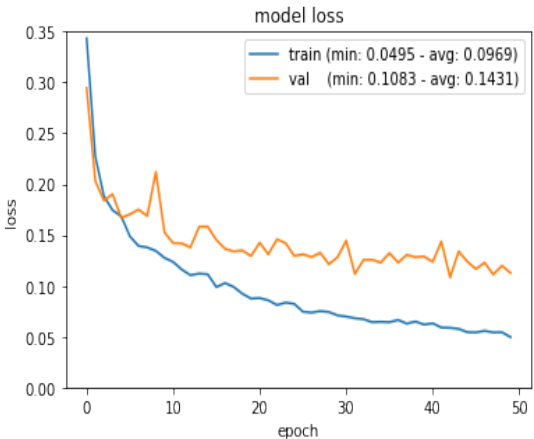
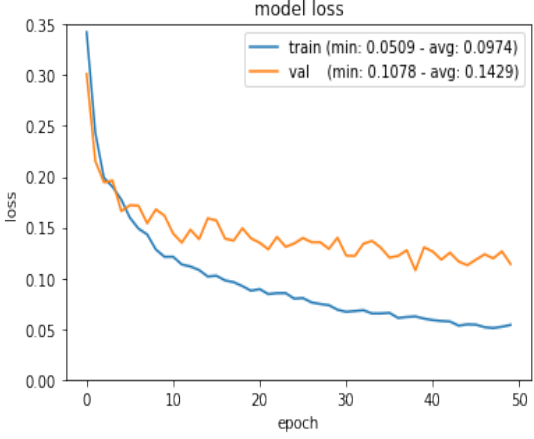
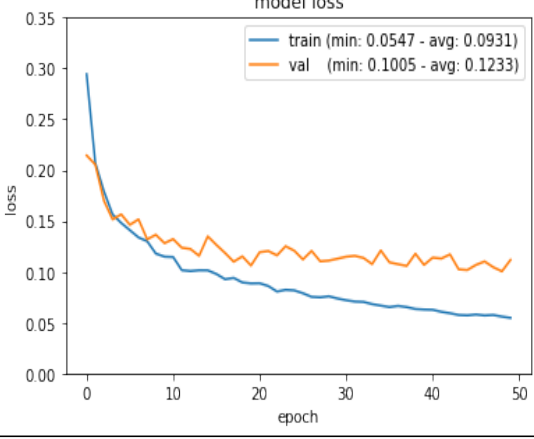
(continuation)

Test	Hyper-parameters	Train Loss
08	EPOCHS = 50 VALIDATION_STEPS = 333 BACKBONE_STRIDES = [4, 8, 16, 32, 64, 96, 128] RPN_ANCHOR_SCALES = (32, 64, 128, 256, 512, 768, 1,024) WEIGHT_DECAY = 0.1	
09	EPOCHS = 50 VALIDATION_STEPS = 333 BACKBONE_STRIDES = [4, 8, 16, 32, 64, 96, 128] RPN_ANCHOR_SCALES = (32, 64, 128, 256, 512, 768, 1,024) WEIGHT_DECAY = 0.5	
10	EPOCHS = 50 VALIDATION_STEPS = 333 BACKBONE_STRIDES = [4, 8, 16, 32, 64, 96, 128] RPN_ANCHOR_SCALES = (32, 64, 128, 256, 512, 768, 1,024) WEIGHT_DECAY = 0.5 TRAIN_BN = True	

(continue)

Table 16 – Hyper-parameters tested in Mask R-CNN tattoo location.

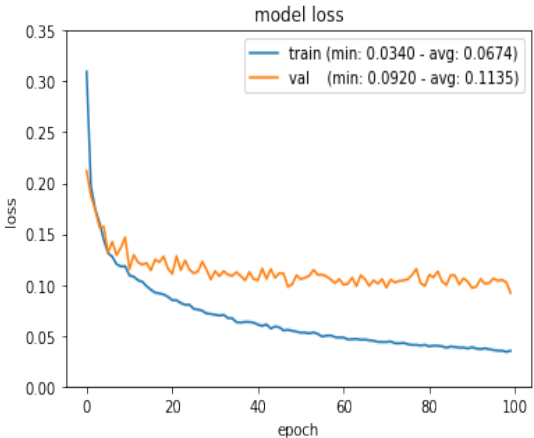
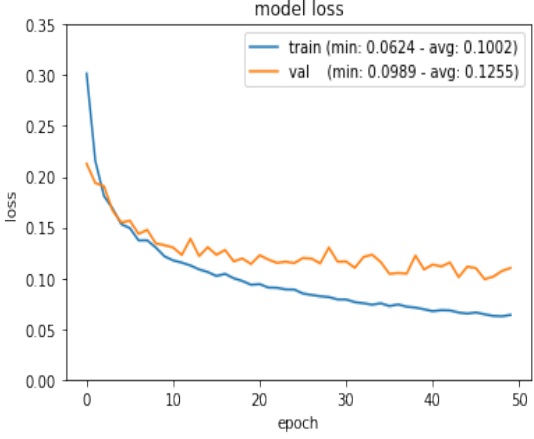
(continuation)

Test	Hyper-parameters	Train Loss
11	EPOCHS = 50 VALIDATION_STEPS = 333 BACKBONE_STRIDES = [4, 8, 16, 32, 64, 96, 128] RPN_ANCHOR_SCALES = (32, 64, 128, 256, 512, 768, 1,024) WEIGHT_DECAY = 0.5 TRAIN_BN = True ROI_POSITIVE_RATIO = 0.66 TRAIN_ROIS_PER_IMAGE = 512 DETECTION_NMS_THRESHOLD = 0.6	
12	EPOCHS = 50 VALIDATION_STEPS = 333 BACKBONE_STRIDES = [4, 8, 16, 32, 64, 96, 128] RPN_ANCHOR_SCALES = (32, 64, 128, 256, 512, 768, 1,024) WEIGHT_DECAY = 0.5 TRAIN_BN = True ROI_POSITIVE_RATIO = 0.75 TRAIN_ROIS_PER_IMAGE = 512 DETECTION_NMS_THRESHOLD = 0.7	
13	EPOCHS = 50 VALIDATION_STEPS = 407 BACKBONE_STRIDES = [4, 8, 16, 32, 64, 96, 128] RPN_ANCHOR_SCALES = (32, 64, 128, 256, 512, 768, 1,024) WEIGHT_DECAY = 0.5 TRAIN_BN = True ROI_POSITIVE_RATIO = 0.75 TRAIN_ROIS_PER_IMAGE = 512 DETECTION_NMS_THRESHOLD = 0.7	

(continue)

Table 16 – Hyper-parameters tested in Mask R-CNN tattoo location.

(continuation)

Test	Hyper-parameters	Train Loss
14	EPOCHS = 100 VALIDATION_STEPS = 407 BACKBONE_STRIDES = [4, 8, 16, 32, 64, 96, 128] RPN_ANCHOR_SCALES = (32, 64, 128, 256, 512, 768, 1,024) WEIGHT_DECAY = 0.5 TRAIN_BN = True ROI_POSITIVE_RATIO = 0.7 TRAIN_ROIS_PER_IMAGE = 512 DETECTION_NMS_THRESHOLD = 0.7	
15	EPOCHS = 50 VALIDATION_STEPS = 407 BACKBONE_STRIDES = [4, 8, 16, 32, 64, 96, 128] RPN_ANCHOR_SCALES = (32, 64, 128, 256, 512, 768, 1,024) WEIGHT_DECAY = 0.5 TRAIN_BN = True ROI_POSITIVE_RATIO = 0.7 TRAIN_ROIS_PER_IMAGE = 512 DETECTION_NMS_THRESHOLD = 0.7 IMAGE_MIN_DIM = 128 MAX_GT_INSTANCES = 120 DETECTION_MAX_INSTANCES = 120	

Source: own author.

APPENDIX B – MASK R-CNN DEFAULT PARAMETERS

Table 17 – Mask R-CNN Default Parameters.

Parameter	Default Value
BACKBONE	resnet101
BACKBONE_STRIDES	[4, 8, 16, 32, 64]
BATCH_SIZE	2
BBOX_STD_DEV	[0.1 0.1 0.2 0.2]
COMPUTE_BACKBONE_SHAPE	None
DETECTION_MAX_INSTANCES	100
DETECTION_MIN_CONFIDENCE	0.7
DETECTION_NMS_THRESHOLD	0.3
FPN_CLASSIF_FC_LAYERS_SIZE	1024
GRADIENT_CLIP_NORM	5.0
IMAGES_PER_GPU	2
IMAGE_CHANNEL_COUNT	3
IMAGE_MAX_DIM	1024
IMAGE_META_SIZE	14
IMAGE_MIN_DIM	800
IMAGE_MIN_SCALE	0
IMAGE_RESIZE_MODE	square
IMAGE_SHAPE	[1024 1024 3]
LEARNING_MOMENTUM	0.9
LEARNING_RATE	0.001
LOSS_WEIGHTS	'rpn_class_loss': 1.0, 'rpn_bbox_loss': 1.0, 'mrcnn_class_loss': 1.0, 'mrcnn_bbox_loss': 1.0, 'mrcnn_mask_loss': 1.0
MASK_POOL_SIZE	14
MASK_SHAPE	[28, 28]
MAX_GT_INSTANCES	100
MEAN_PIXEL	[123.7 116.8 103.9]
MINI_MASK_SHAPE	(56, 56)
POOL_SIZE	7
POST_NMS_ROIS_INFERENCE	1000
POST_NMS_ROIS_TRAINING	2000
PRE_NMS_LIMIT	6000
ROI_POSITIVE_RATIO	0.33
RPN_ANCHOR_RATIOS	[0.5, 1, 2]
RPN_ANCHOR_SCALES	(32, 64, 128, 256, 512)
RPN_ANCHOR_STRIDE	1
RPN_BBOX_STD_DEV	[0.1 0.1 0.2 0.2]
RPN_NMS_THRESHOLD	0.7
RPN_TRAIN_ANCHORS_PER_IMAGE	256
STEPS_PER_EPOCH	1000
TOP_DOWN_PYRAMID_SIZE	256
TRAIN_BN	False
TRAIN_ROIS_PER_IMAGE	200
USE_MINI_MASK	False
USE_RPN_ROIS	True
VALIDATION_STEPS	50
WEIGHT_DECAY	0.0001

Source: own author.

APPENDIX D – RESULTS OF UNKNOWN CLASSES CLASSIFICATION

Table 18 – Results of the classification of unknown classes.

Original	Predicted	Original	Predicted	Original	Predicted	Original	Predicted
baby_01	women face	crown_01	taz	dagger_01	leprechaun	dolphin_01	fish
baby_02	women face	crown_02	heart	dagger_02	spider	dolphin_02	horse
baby_03	man face	crown_03	octopus	dagger_03	horse	dolphin_03	heart
baby_04	man face	crown_04	angel	dagger_04	guitar	dolphin_04	butterfly
baby_05	women face	crown_05	angel	dagger_05	knife	dolphin_05	taz
baby_06	heart	crown_06	butterfly	dagger_06	guitar	dolphin_06	heart
baby_07	man face	crown_07	clown	dagger_07	taz	dolphin_07	wings
baby_08	Jesus	crown_08	guitar	dagger_08	joker	dolphin_08	star
baby_09	women face	crown_09	cat	dagger_09	taz	dolphin_09	bird
baby_10	women face	crown_10	clown	dagger_10	flower	dolphin_10	bird
elephant_01	fish	falcon_01	wings	frog_01	butterfly	gorilla_01	clown
elephant_02	leprechaun	falcon_02	guitar	frog_02	heart	gorilla_02	women face
elephant_03	horse	falcon_03	flower	frog_03	leprechaun	gorilla_03	man face
elephant_04	leprechaun	falcon_04	dog	frog_04	leprechaun	gorilla_04	man face
elephant_05	horse	falcon_05	eagle	frog_05	butterfly	gorilla_05	lion
elephant_06	horse	falcon_06	eagle	frog_06	horse	gorilla_06	Jesus
elephant_07	octopus	falcon_07	eagle	frog_07	butterfly	gorilla_07	yin-yang
elephant_08	man face	falcon_08	eagle	frog_08	flower	gorilla_08	Jesus
elephant_09	eagle	falcon_09	eagle	frog_09	octopus	gorilla_09	man face
elephant_10	horse	falcon_10	wings	frog_10	butterfly	gorilla_10	man face
mandala_01	dragon	moon_01	yin-yang	panter_01	horse	shark_01	eagle
mandala_02	demon	moon_02	yin-yang	panter_02	horse	shark_02	taz
mandala_03	taz	moon_03	heart	panter_03	dragon	shark_03	taz
mandala_04	spider web	moon_04	heart	panter_04	dog	shark_04	snake
mandala_05	yin-yang	moon_05	dragon	panter_05	flower	shark_05	bird
mandala_06	taz	moon_06	heart	panter_06	horse	shark_06	women face
mandala_07	fish	moon_07	butterfly	panter_07	dog	shark_07	heart
mandala_08	demon	moon_08	heart	panter_08	horse	shark_08	star
mandala_09	spider	moon_09	heart	panter_09	taz	shark_09	guitar
mandala_10	heart	moon_10	heart	panter_10	cat	shark_10	eagle
sun_01	flower	unicorn_01	tiger	witch_01	taz	wolf_01	demon
sun_02	spider	unicorn_02	horse	witch_02	angel	wolf_02	dog
sun_03	spider	unicorn_03	star	witch_03	leprechaun	wolf_03	cat
sun_04	heart	unicorn_04	taz	witch_04	guitar	wolf_04	cat
sun_05	angel	unicorn_05	fish	witch_05	clown	wolf_05	car
sun_06	butterfly	unicorn_06	joker	witch_06	leprechaun	wolf_06	dog
sun_07	spider	unicorn_07	octopus	witch_07	horse	wolf_07	cat
sun_08	taz	unicorn_08	horse	witch_08	guitar	wolf_08	dragon
sun_09	flower	unicorn_09	dragon	witch_09	horse	wolf_09	cat
sun_10	wings	unicorn_10	horse	witch_10	clown	wolf_10	tiger

Source: own author.