UNIVERSITATEA POLITEHNICA BUCUREȘTI
FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
DEPARTAMENTUL CALCULATOARE

PROIECT DE DIPLOMĂ

Detecția imaginilor cu impact emoțional
și clasificarea tipurilor de violență
folosind rețele neurale adânci

Edwin-Mark Grigore

Coordonator științific:
Conf. dr. ing. Traian Rebedea

BUCUREŞTI

2020

UNIVERSITY POLITEHNICA OF BUCHAREST
FACULTY OF AUTOMATIC CONTROL AND COMPUTERS
COMPUTER SCIENCE DEPARTMENT

DIPLOMA PROJECT

Graphic Content Detection in Images and Classification of Types of Violence Using Deep Neural Networks

Edwin-Mark Grigore

Thesis advisor:
Conf. dr. ing. Traian Rebedea

BUCHAREST

2020

CONTENT

SINOPSIS

Copiii sunt cel mai ușor de influențat. Fie că acest lucru se face prin cuvinte sau prin fapte, prin ceea ce văd sau prin ceea ce aud, ei sunt la vârsta la care cel mai mic lucru își poate lăsa amprenta asupra lor. O imagine nepotrivită ajunsă sub ochii unui copil de 5 ani îl poate traumatiza pe viață sau, cel puțin, îi poate genera o temere de care să nu poată scăpa niciodată.

Tehnologia a evoluat, s-a extins într-atât de mult, încât este prezentă în viețile tuturor în cele mai mici detalii, și din ce în ce mai pregnant, la îndemâna copiilor. Pornind de la această realitate este necesară detecția imaginilor ce conțin scene de violență sau scene cu impact emoțional puternic, imagini ce conțin sânge sau înfățișează corpuri umane având plăgi deschise sau cadavre, incendii violente sau prezența armelor.

Modelele de învățarea automată sunt capabile să extragă caracteristicile imaginilor și să învețe să identifice imaginile care înfățișează scene nepotrivite pentru copii, folosind diferite tehnici: fie prin metode ML tradiționale, fie folosind SVN, fie rețele neurale adânci. Puținele modele existente ce se adresează acestei probleme pot atinge performanțe de până la 90% precizie.

ABSTRACT

The children are the easiest to influence. Whether this is done by words or by actions, through what they see or what they hear, they find themselves at the age when the smallest thing can leave a mark on them. An inappropriate image that comes under the attention of a 5-year-old child, can traumatize him for life, or at least cause him severe fear that he would never be able to get rid of.

The technology has evolved and expanded so much over the last decades, that is present in everybody's life in every little aspect, and more and more significantly at children's disposal. Starting from this reality, it is necessary the identification of the images that contain scenes of violence or emotionally disturbing scenes, images that contain blood or depicts human bodies with open wounds, violent fires or presence of guns and weapons.

Machine learning is capable of extracting features from images and learn to identify the images that depicts inappropriate scenes for children, using different techniques: either through traditional ML methods, or using SVN or deep neural networks. The few existing models that address this problem can reach performances of up to 90% accuracy.

1 INTRODUCTION

1.1 Background

1.1.1 Children Exposure to Violence

The digital era has allowed most people of the planet to be connected to the Internet via continuously shrinking devices: personal computers, smartphones, smart gadgets, smartwatches etc., meaning anyone has access to any kind of information at any moment. Alongside the obvious and huge advantages Internet, media and technology altogether have brought to the world, there are many risks worth noting and addressing.

Focusing on the rapid spread of smart gadgets in the homes of most of the people, forth comes an issue that has been long debated and argued: children exposure to violence through media. Whether is television or movies, games, news, everything a child hears and perceives has an impact on him/her. It is in the hands of the parents to prevent their children from seeing images that would cause them anxiety, fear, misunderstanding of society, or even traumas.

The common adage "A picture is worth a thousand words" denotes exactly how an image can influence a child, especially if we are talking about inappropriate images. Studies [1] have shown that aggressive or antisocial behaviour is heightened in children after watching violent television or films. Science tells us that early exposure to extremely fearful events affects the developing brain, particularly in those areas involved in emotions and learning [2]. When children see images that are emotionally disturbing, images that depicts the world in an inadequate manner for their young minds to comprehend, they can learn fear from situations they should not be exposed to. The brain stores what we see, and how often do we hear people say, "I have seen something I will not forget my entire life". If that is the case for mature persons, then even more importance should be given to what children are allowed to see.

1.1.2 Parental Control Applications

Most of the online media platforms that are available to children and have no age restrictions by default – i.e., film-producer or film-provider platforms, video-sharing platforms, applications stores, online electronic bookstores etc. – offer mechanisms of protection targeted on age, mainly named *Parental Control*, that will allow the parents to set restrictions on the content their children must not access.

Recently, there have been developed several parental control applications by different companies in order to aid parents overcome the issues the ever-growing technology comes along with. Many give the possibility to the parents to track their children's both online activity (i.e.

accessed websites) and offline activity (e.g. applications used, time spent on devices etc.), as well as to keep a record of the people their children interact with via calls or chats.

The potential dangers are identified through complex artificial intelligence algorithms that would analyse and detect any kind of inappropriate content on their devices, whether it is an image sent in a chat or a bullying text message received, subsequently alerting the parents about the problem and blocking the threat.

This way, the children are protected from being exposed to any harmful content or situation they may encounter while browsing the Internet or chatting with others.

1.1.3 Classification of violence and graphic content

The term *violence* is "used to animal and human behaviour that threatens to cause or causes severe harm to a target" [3].

Violence is "the intentional use of physical force or power, threatened or actual, against oneself, another person, or against a group or community, which either results in or has a high likelihood of resulting in injury, death, psychological harm, maldevelopment, or deprivation" [4].

There has been much controversy about this term and about what should be included under the heading of violence. The World Health Organization divided violence into three categories: self-directed violence, interpersonal violence, and collective violence [4]. These three categories are divided further to depict more specific types of violence: physical, sexual, psychological, emotional.

Transposing the notions to the field of images, the violence transcends these categories. The term *graphic* or *graphic violence* refer to depiction acts of violence in visual media such as film, television and video games. The violence may be real or simulated.

Graphic violence generally consists of uncensored depiction of various violent acts which includes depiction of murder, assault with a deadly weapon, accidents which result in death or severe injury, and torture.

Graphic violence causes intense emotions, especially in children, and frequent exposure can lead to desensitization over the severity of the actions depicted. Over time, children may become too familiar with violence and may learn that a little violence is socially acceptable, mainly because it is present in many contexts, such as films, news, literature, that children are accustomed to and take as a natural thing.

Many online media platforms have defined standards and policies regarding this subject, and some have restricted, or even prohibited inappropriate content on their platforms. Social-media

platforms (e.g. Facebook) have introduced the guidelines for their users, which act as rules to be followed when interacting with other users.

Facebook wrote the *Community Standards* [5] which stipulates the rules that platform users should take into consideration when posting. On this basis, they promote and fight to maintain a "safe environment on Facebook" by removing any content that is identified as graphic, upsetting or harmful for viewers, depicting different kinds of violence in images, videos or speech. Facebook divides prohibited content into several categories, starting with hate speech, listed as a direct attack based on race, ethnicity, religious affiliation, sexual orientation, gender or disability and all the way to nudity and sexual activity, violence and criminal behaviour, which include images or videos of people or dead bodies in non-medical settings, child or adult abuse, live streams of capital punishment of a person, still images depicting the violent death of a person. Facebook have their own AI algorithms that detects such images and videos and moderators that apply the rules, by censoring and removing posts that do not follow the rules.

In this context, this project would like to address this issue with technology in respect of its impact on children and their healthy emotional and psychological growth.

1.2 Problem Description

In order to prevent the exposure of children to graphic and violent images, these images must be identified first. Since parents cannot be physically near their children every single time, they must rely on the technology they use to do that.

As the technology advances, Computer Vision leads the way in training artificial intelligence to learn to interpret and understand the visual world. Using deep learning models, we now have machines that can accurately identify and classify objects. Computers can acquire and process data, analyse and understand the content of the images, and they can extract the information necessary to perform different tasks and decide based on the content of the image.

Although there is extensive knowledge to develop such deep learning models, only a few have been created that recognize and/or classify the violence depicted in still images, and even less are available to the public use.

There are several potential areas that may use machine learning to detect violence, such as parental control applications and web filtering, that makes this subject worth being studied.

The human brain can perceive an image and understand the content of the image in the blink of an eye. A team of neuroscientists from MIT found that the brain can process an image even if the eyes see it for only 13 milliseconds [6]. They used rapid serial visual presentation (RSVP) of series of 6 to 12 pictures presented at between 13 to 80 milliseconds per picture. The participants were

asked to detect a picture specified by name, which they did. The accuracy of detection improved with increasing the duration of presentation, nevertheless the result of the study shows how fast humans can understand the meaning of a picture.

Judging an image whether it contains violence or not it is slightly more difficult. While most of the time, one can do it relatively fast, some situations take a little longer for an answer to be given.

Assigning this job to a computer is no easy task. Computer Vision includes several algorithms that will successfully identify objects in images with almost 100% accuracy. However, identifying the graphic content is more than simply object identification.

The depiction of violence is relative. The brain perception of two persons is different, so one may judge an image as being graphic, while the other may not. Some images that depict the exact same objects, but varying in pose and instance, may convey violence in some cases, but no violence at all in other cases, depending on the context in which those particular objects are depicted. An example, and maybe the most difficult to work with, is fire. An explosion can easily be classified as graphic content, but a fire camp might be mistakenly classified as being upsetting for viewers due to its size or presence of people around it.

The difficulty of the problem comes from this aspect. The context depicted in the images or the circumstances under which the photograph appear to have been taken may be misleading for the artificial intelligence, while for the human eye it would be an easy decision.

1.3 Goals

This project has the following goals:

- Gather a dataset of images that depict the best graphic content and violence
- Train a deep model to recognize and classify the types of violence in still images

1.3.1 Dataset

The purpose of this stage is to gather as many useful sets of data that can be used to accurately train the model. The images must be labelled and must portray the type of violence labelled precisely, clearly. The dataset should not contain any images that are inaccurate.

The dataset can be acquired online downloading from a reliable source or by manually browsing the search-engines and downloading images. The goal is to find a dataset that has been already created, because that will save a lot of time. Ultimately and as a last resort, the second option should be taken into consideration.

After acquiring the dataset, we want to go through the directories and prune irrelevant images (e.g. images that can be double classified, over- or underexposed images where the subject is not visible enough). The trickiness of this step comes from the fact that multiple images can be classified both violent and non-violent by human intelligence, depending on who is making the call and based on the context. This kind of images must be carefully separated into the corresponding labels, so an image that does not show a violent or graphic content, but is similar in composition to one which does, would not end up being mislabelled.

At the end of this step, the dataset should be ready for use and it should consist of as many images as possible. A solid start would be 1000 images per category.

Given the fact that the dataset might be small, despite the efforts, the next step is to extend the size of the dataset. The augmented images are acquired by performing a series of transformation on the base images, that include rotation, cropping, skewing, zooming, flipping and mixing.

Because neural networks learn the features of the images, these features must be represented in as many ways as possible for the trained model to be able to recognize these features in images that it will be used on.

The augmentation can solve numerous problems of the dataset. It does not only increase the size of the dataset considerably, but it can also bring in diversity given by the subtle variation in orientation, position, angle, rotation etc. of the subject depicted. This will result in the dataset to contain, for example, a specific weapon in many positions on an image, increasing the accuracy of the model to be trained.

At the end of this step, the dataset size should increase several times and should depict a diversity of features.

1.3.2 Training the Model

The final goal of the project is the training of the model. At this moment we have a good dataset, which has been augmented and ready to use. This dataset must be used to train a model to recognize violence and graphic scenes in images with a high accuracy.

Transfer learning is used in order to achieve the goal. This method applies different existing models that have already been trained for general purposes, to the characteristics of this project.

Transfer learning can save a lot of training time, given the fact that we start from something that already exists and has already been trained on actual data. Most of the times, the machine learning model that is used has been trained on datasets much larger and complex than our dataset and for a longer time than we can do for our own model. A broad dataset is crucial in feature learning and this will be the case for this project.

There are several steps that must be taken in order to train the model. First, we must choose from the large pool of the existing deep learning models one to be the basis to our training. The process of identification of the model that works best on the dataset available is believed to be a key aspect. Second, we take the pre-trained model and use it as a starting point for our model. Also, we must decide which layers of the pre-trained model that we will use in the process and what layers be must build on top of it. Finally, we must adapt and refine the pair of so it may fit as well as possible the task at hand, process called tuning the model.

1.4 Outline of the Thesis

This thesis presents a project with the goal of creating a machine learning model that can identify and classify violence and other graphic content in still images. The output model will be able to identify the followings: presence of weapons, presences of fire, fight scenes, presence of blood and gore, detailed in Chapter 2. The Chapter 3 will discuss the existing machine learning models, the current technology and alternatives to this project. Then, Chapter 4 describes the solution this project proposes, whose implementation will be described step by step in Chapter 5, and finally, the results will be analysed, and conclusions will be drawn in Chapter 6.

2 PROJECT SPECIFICATIONS

The project aims to provide a solution to identifying potential harming and upsetting images for children up to 14 years old.

The main features the output machine learning model of this project intends to identify and classify are presented in the following lines.

Presence of firearms – the presence of any type of gun or similar fire weapon, whether it is shooting or not, pointed at someone, or threatening a person, regardless of the intent of the subject depicted, is to be classified as violent image.

Firearms visible

Pistol

Gun threatening

Figure 1. Firearms classification[1]

Presence of cold weapons – any type of melee weapon, ranged weapon or other type of weapon that does not involve fire or combustion, is to be classified as violent image.

Sword

Sword fight scene

Knife threatening

Figure 2. Cold weapons classification1

[1] See Appendix 2 for image sources.

Presence of fire – any explosion caused by a bomb, any large-scale fire, vegetation fire, any human or animal, living or dead that is burning, any fire caused by a gun, is to be classified as violent.

Explosion

Vegetation fire

Human on fire

Figure 3. Fire classification[2]

Fight scenes – any image that represent a fight, regardless of the number of people involved or how they are fighting or the weapons they use, is to be classified as violent image. A fight scene may imply punching, kicking, mutual or from one side. Battle scenes struggles between a person and an animal will also be included in this category.

Sports

Street fight

War

Figure 4. Fight scenes classification[2]

Presence of blood and gory scenes – any serious body injury, any presence of blood that drains out from a body, any wound or tissue damage, any dead body that shows significant injury, presence of horror creatures, mutant creatures or skull and flesh representation, is to be classified as violent image.

[2] See Appendix 2 for image sources.

Head injury

Dead body

Mutant creature

Figure 5. Gore classification[3]

Any other image that does not contain violent depictions or graphic contents, and that is not classified under the above categories, will be labelled as ***non-violent***.

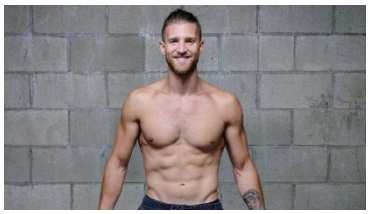


Figure 6. Non-violent images classification[3]

The prediction through the model must be made a straight-forward task by wrapping the actions that are necessary in order to classify an image in a script that must be easy to use.

The model may be subsequently integrated into a parental control application, providing the means to detect if the images a user opens, receives or sends, are graphic or contain violence and notify the parents about the potential harmful situation.

[3] See Appendix 2 for image sources.

3 STATE OF THE ART

Deep learning has seen an immense increase in interest over the last few years. Since the term *deep learning* was introduced in 1986 by Rina Dechter [7] to the machine learning community, many steps have been taken to elaborate and build the architecture we know today. Deep learning has since been applied to fields including computer vision [8][9], speech and audio recognition, social network filtering, many producing results compared to or even surpassing the human ability [10].

It is called *deep* because the architecture uses multiple layers in the network. The unbounded number of layers allows an optimized implementation and has practical application.

The countless work that has been done in this field is great and listing all the research that have brought deep learning and computer vision to what it is today is beyond the purpose of this paper. One of them, however, is worth noting.

Believing that, despite all the hard work, research and discoveries made, something was still missing, the Stanford University professor, Fei-Fei Li launched in 2009 the ImageNet [11]. It was something that had not been done until that time. Fei-Fei Li thought that the missing part was the data. And while there were images all over the Internet, there was no such thing as a database containing labelled images. Therefore, she assembled a free database consisting of more than 14 million labelled images, saying, “Our vision was that Big Data would change the way machine learning works. Data drives learning” [42]. They used the dataset in the annual Large-Scale Visual Recognition Challenge (LSVRC) where contestants would build computer vision models which were rated by their accuracy.

In 2012, Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton submitted a model that was almost two times better than the previous ones, with an error rate of 15.3%. They used GPUs to train the model, as their power of computing offered a great amount of speed. They also reduced overfitting by introducing the method called *dropout* and used the *rectified linear activation unit* (ReLU), methods that later became the basics to deep learning. The network is known as *AlexNet* [9].

Since then, computer vision and deep learning have seen a rise in interest, research and applications. Deep learning is now present almost in any domain of activity: defeating the world best player at Go [10], AI-driven cars [12], diagnosing skin cancer [13] or surveillance identification, only to name a few.

3.1 Violent Images Datasets

Since ImageNet, several images datasets have been assembled and are available for free. They gather and label images for thousands of categories, each consisting of hundreds of images. The categories of these datasets are various, so the dataset can train a model more precisely in order to serve a broad purpose in visual recognition.

According to specific needs of different projects, many smaller datasets have been created, each consisting of images from different categories that depict the subject those models should recognise, which would allow them to train a network over an existing one that has already been trained with a huge dataset such as ImageNet.

For this purpose, some datasets of guns and weapons images that have been used to train models that can detect weapons in images or videos can be found on the Internet[4] and are free to use [18].

However, the number of datasets of images containing fight scenes or gory scenes is low, and even those are very small and not concluding. A slightly more convincing number of datasets can be found for videos depicting violence, but still, when it comes to fights or blood presence, there are not too many.

Since dataset that focus on general activities cannot be used for this project, and since a collection with violence-depicting images does not exists, this problem must be approached in a different way.

One way is to start from a video dataset and work on it. This would require a lot of processing to extract images, to select the ones with the best feature display etc. Such video datasets can be found online more easily. RWF-2000 [19] and VSD from InterDigital [14] are only two examples of many.

Another way is to start from scratch and to build a new dataset with images downloaded from online sources (e.g. Google, Bing), with each image labelled afterwards.

Both methods are time consuming, one for the processing part and the other for the searching part. The situation is not ideal, so perhaps a better option is to combine the methods and save some time from both. The resulting dataset would be good enough, and hopefully, large enough.

[4] https://sci2s.ugr.es/weapons-detection, https://github.com/SasankYadati/Guns-Dataset, http://kt.agh.edu.pl/matiolanski/KnivesImagesDatabase/

3.2 Pretrained Image Recognition Models

Pretrained models are widely used in AI recognition. When building a model from zero is not possible due to time restrictions, computational or dataset limitations, pretrained models are a viable option, offering instant efficient results.

Since ImageNet and AlexNet, several models have been trained with large amount of data for general image recognition, whether is object detection, face detection or action detection. Learning the secrets of the technology allowed the researchers to create more sophisticated networks with an increasing number of layers in order to reduce the error. Thus, the dept of the neural network has revolutionised the technology.

AlexNet has 8 convolutional layers [9], the VGG network has 19 [22], Inception has 22 [23], while ResNet has 152 layers [24]. ResNet-50 is a smaller version of ResNet 152 and is widely used for transfer learning. Some of these pretrained models have been used in the implementation of this project.

3.3 Violence Detection Models

A quick research online would reveal a scarce in machine learning model specialized in detecting violence in images. There are several models trained to detect violence in videos, many of them being used in surveillance systems.

O. Deniz et al. introduced a *fast violence detection in video* in 2014 at the International Conference on Computer Vision Theory and Applications. They claim to have improved the methods of recognition to the point of making it 15 times faster than other models released at that time, also with an accuracy rate increased by 12% [20].

Peipei Zhou et al. proposed in 2018 a model that detects violence in surveillance videos using low-level features to represent the appearance and the dynamics for violent behaviour [21].

However, regarding violence detection in images there has been less work done over the years. Many of the models implemented so far focus on generic human action recognition (e.g. running, walking, playing sports), rather than specific actions (in this case, fighting, threatening with a weapon, injuring a person etc.).

Perhaps, several reasons for this lack of work in this aspect can be found: (1) videos are composed of sequences of still images at high rates per second, therefore, video detection is based on image detection, up to a certain point; (2) there are not so many practical uses of violence detection in images etc. Nevertheless, I believe that contributing to this field of work can bring advantages to the community and to the world. Some immediate application can be, as mentioned before, child protection, or posts filtering on social media, to name a few.

Addressing the latter, Facebook continuously strive to be on top by always developing and enhancing their AI to track and remove graphic content and other questionable content from their platform. In 2018, Facebook revealed a part of their ways of using AI in combating these issues to the public: frameworks, tools, libraries and models. They are available at https://ai.facebook.com/tools.

3.4 Integrated Technologies

The implementation of this project involved the use of multiple modules and existing technology. These modules are presented in Table 1 and described in the following lines.

Table 1. Technology used for implementing the project

Technology	Name	Description
Libraries & APIs	TensorFlow	Open-source platform used for machine learning applications such as neural networks.
	Keras	Open-source neural-network library for Python.
	OpenCV	Open source computer vision and machine learning software library.
Datasets	VSD	Public dataset for detection of violent scenes in videos created by InterDigital.
	Google Images	Used for manual collection of images.
Pre-trained models	ResNet-50	Convolutional neural network, the smaller version of ResNet-152
	VGG16	Convolutional neural network with 16 weight layers.

3.4.1 TensorFlow and Keras

TensorFlow is an open-source platform for machine learning. It is a symbolic math library used for many machine learning applications. It was developed by the Google Brains Team and released in 2015, before being used internally by Google [25].

The platform offers different levels of abstraction, providing flexibility and intuition in implementing. It allows you to build and train a model with any level of experience, from starters to experts. The Keras API helps in the process by making the integration of the architecture easy.

Keras is TensorFlow's high-level API for building and training deep learning models [26]. It is used for prototyping, research and production. Being easy to use is perhaps the greatest advantage, with simple and clear interface. Keras is used in research by CERN, NASA etc.

TensorFlow is available at www.tensorflow.org. Keras API documentation is available at www.keras.io.

3.4.2 Violent Scene Dataset by InterDigital

Violent Scene Dataset (VSD) created by InterDigital [14-17] is a public dataset for the detection of violent scenes in videos. It is a collection of labels, features and annotations based on the extraction of violent scenes from films and web videos. It also contains audio annotations of violence-depicting sounds present in the videos. According to the description made in the publications [15], the data was produced by the Fudan University and the Ho Chi Minh University of Science.

The dataset consists of 86 short videos downloaded from YouTube and normalised to a frame rate of 25. Also, the dataset contains ground truth created from a collection of 32 films of different genres (which are not included in the dataset due to copyright issues).

The violence identification is made based on two definitions of violent scenes: (1) subjective definition and (2) objective definition. The subjective definition describes a violent scene as a "scene one would not let an 8-year-old child see because they contain physical violence"[5]. The objective definition shows that a violent scene contains "physical violence or accident resulting in human injury or pain"[6].

The dataset is available by request on the InterDigital website, www.interdigital.com.

3.4.3 Pre-trained Models

ResNet (Residual Networks) is a neural network used in a variety of computer vision applications. It was first introduced in 2015 at ImageNet contest by He et al. [24] and won the competition scoring the lowest classification error: 3.57%. The residual networks are easier to optimize, according to Kaiming He. For the ImageNet contest, residual networks of 152 layers have been

[5] As presented in the description of the dataset. https://www.interdigital.com/data_sets/violent-scenes-dataset
[6] See previous note.

used, that is 8 times deeper than existing VGG model, preserving at the same time a lower complexity [37].

The weights and dataset are available at https://www.kaggle.com/keras/resnet50/home.

VGG is a neural network introduced by Simonyan and Zisserman [22] in 2014. The model scored 92.7% accuracy in ImageNet contest the same year. It replaces the large filters in AlexNet with multiple 3x3 filters in sequence [41]. VGG16 was trained on the ImageNet dataset. At that time, a neural network of 16 or 19 layers was considered deep enough. The model is used in many deep learning image classification tasks.

4 METHOD

The project presents a solution to the problem of detecting the violence and any kind of graphic content in still images. The goal of the project, as mentioned in Chapter 1, is to train a deep learning model that would recognise the types of violence presented in Chapter 2, i.e. presence of weapons, presence of fire, presence of blood, fight scenes. Similar to the work previously done to create state-of-the-art models, in order to achieve this goal, four crucial steps (see Figure 7) must be taken: (1) identify the violence sources in images and settle on what categories the model should be trained to recognise, (2) create a viable dataset for detecting violence, (3) choose the appropriate pre-trained model that would ensure the best results in terms of accuracy, space and time consumption, (4) tune and train the model in order to find the best solution.

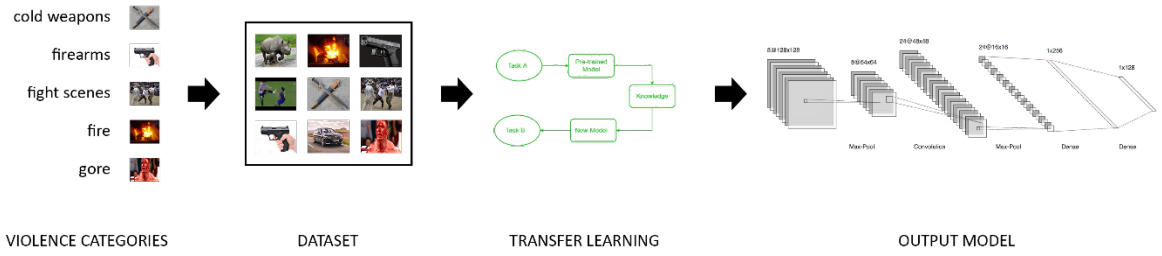


Figure 7. Steps to creating the model[7]

The first step consists of a little research needed to be done in order to decide on the categories that fall under the meaning of violence. Here, it is necessary to learn what images can have an emotional impact on the viewers, especially children, and how to distinguish the images depicting violence from those that do not. More details in Section 4.1.

The second step is to acquire a set of images that would meet the requirements as training and validation sets. Online searches should result in some datasets which can be used at least as a starting point for compiling a proper dataset. More details in Section 4.2.

The third step is represented by the choice of pre-trained model. Various models are available (e.g. ResNet-50, Inception V3, Xception, VGG19 etc.), but choosing one that fits the needs of the project is important in order to achieve good results. More details in Section 4.4.

[7] See Appendix 2 for image source.

The fourth step implies using the transfer learning method to build the final model on top of the pre-trained model and apply fine-tuning in order to improve the accuracy of the model. More details in Section 4.3.

Finally, the model can be wrapped up under a small script that would receive the input image and would return the classification results.

4.1 Violence Classification

As debated earlier in this paper, the problem of identifying the violence depicted in still images might be a challenge. Many factors such as cultural opinion, perspective, intuition and prior exposure to violence can influence a person in deciding whether the content of an image is harmful or graphic, or not.

However, a general list of categories that are widely recognized as violence picturing can be made, containing the following, as stated in Chapter 2:

- Violence depicted through weapons
    - Cold weapons
    - Firearms
- Violence depicted in human or animal bodies representing wounds, flesh, and any kind of gory scene
- Violence depicted under the form of isolated or group fights, with weapons or bare hands
- Violence depicted through fire, i.e. wild vegetation fire, buildings on fire, bodies on fire, explosions, or any large fire

An image that does not fall under any of the categories above will, therefore, be classified as non-violent.

This list can be undoubtedly improved, by adding elements such as car accidents or any other vehicle crashes as a potential violence source. But due to lack of concrete resources available in terms of sets of images and correct interpretation of the boundary which divides violent and non-violent depiction in these cases, I opted against the integration of this category. Future work on this project will see this category added on the list.

4.2 Building the dataset

Long searches on the available platforms resulted in disappointing findings. Except for small datasets on handguns and rifles, there was nothing much I was able to find. In consequence, I

made the choice to use video datasets and work on them, as there were more of the kind available.

I opted for the VSD dataset from InterDigital based on the amount of data they offer and how well organized it is. The categories of violence of the datasets correspond to the ones of this project, thus making it easy to work with.

This VSD dataset contains 86 videos that are both video and audio annotated for violence. In addition, 31 Hollywood films have also been processed, and the ground truth files are contained in the dataset, alongside the sources used to obtain the films.

4.2.1 Frames Extraction

Due to the fact that videos are delivered, not images, processing work needs to be done. Each frame must be extracted from the video, sorted according to the annotation and saved into the new database I created. For each label, a separated folder is created. At the end of this step, manual inspection of the resulting set of images is required. Duplicate images should be removed, images that are blurry, darkened or where the subject is unclear should be removed, also mislabelled images should be moved to the proper category, or removed if necessary. Optional, all images can be resized to the specific size required by the model.

4.2.2 Google Images

The number of images resulted in the process of extraction is in the tens of thousands. However, after the thorough and perhaps overzealous manual inspection and repeated deletion of the unusable files, the database consisted of only around 1000 images, which is too little for a machine learning training set. Also, different perspectives on different categories failed to be gathered into the database, especially for gore and fire presence. At this point, I started to search on Google images that would picture the situations that were missing, also in order to extend the dataset.

4.3 Transfer Learning

Transfer learning is a method of machine learning which allows to repurpose a model trained on one task. The main advantage that transfer learning offers is speed. Modelling the second task can be made with improved performance and a lot easier using the model that was first trained for a general task, learning general features. [29].

"Transfer learning is the improvement of learning in a new task through the transfer of knowledge from a related task that has already been learned." – Chapter 11: Transfer Learning, Handbook of Research on Machine Learning Applications [30].

While traditional machine learning implies single task learning and isolated training with no influence over the knowledge that is to be learned whatsoever, in transfer learning one can use knowledge acquired from previously trained model (i.e. features, weights) to train a new model on a smaller dataset that is compatible with the one that is the source of the knowledge for the base model [31]. A comparison between the two methods can be observed in Figure 8.

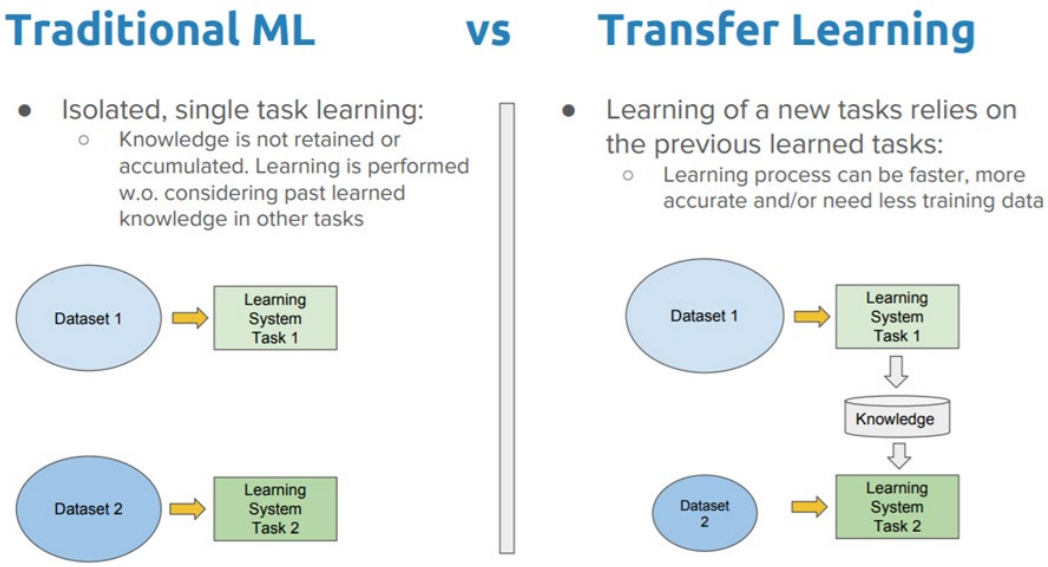


Figure 8. Traditional learning vs Transfer learning[8]

There are three main aspects that must be addressed in the process of transfer learning. First, we need to identify what part of the knowledge can be transferred from the source task to the target in order to improve the performance and, also, we need to understand what the common things between the tasks are. Then, we must choose carefully when to utilize the method, as there can be tasks where applying transfer learning would not increase the performance at all, but on the contrary. Finally, once we decide what we need and that transfer learning is useful on the task, we must find the ways to transfer the knowledge from the source task to the target task and choose the best strategy to be applied [31].

[8] See Appendix 2 for image source.

There are different strategies to approach the transfer learning method. Based on the characteristics of the domains, these strategies can be divided into three categories [31] as presented in Table 2:

- Inductive Transfer Learning
- Unsupervised Transfer Learning
- Transductive Transfer Learning

Inductive Transfer Learning implies that both the source and the target are the same, but the tasks differ. The algorithm will use the inductive bias of the source domain to improve the results of the target task [31].

Unsupervised Transfer Learning is similar to Inductive Transfer Learning, but concentrating on unsupervised tasks in the target domain, which differ from the source task. No labels are available [31].

Transductive Transfer Learning is used for different corresponding domains. The source has labeled data, but the target task has no labels [31].

Table 2. Transfer Learning Strategies [31]

Strategy	Domains	Labels (source)	Labels (target)	Tasks	Tasks Type
Inductive	Same	Un- / Available	Available	Related	Regression, Classification
Unsupervised	Related	Unavailable	Unavailable	Related	Clustering, Dimensionality Reduction
Transductive	Related	Available	Unavailable	Same	Regression, Classification

To understand what to transfer between the tasks, several approaches have been proposed: *instance transfer* [32] (when the source domain cannot be used directly, using only specific parts from it along with the target data will improve the performances), *feature-representation transfer* [34] (identifies good cross-domain feature representations and reduces the domain divergence and the error rates), *parameter transfer* (assumes that related tasks share parameters or hyperparameters distribution) etc.

4.3.1 Deep Learning

Deep learning is the representation of the inductive learning strategy. Algorithms of this type are used to deduce a correspondence from a training set. In this case, violence classification, the model would learn to match the input features of an image to the correspondent class label. The notion of *inductive bias* helps the algorithms to perform better on unseen data, by making assumptions on the distribution of the training data. Inductive transfer also utilizes Bayesian and Hierarchical Transfer in the process of training the target task (Figure 9) [31].

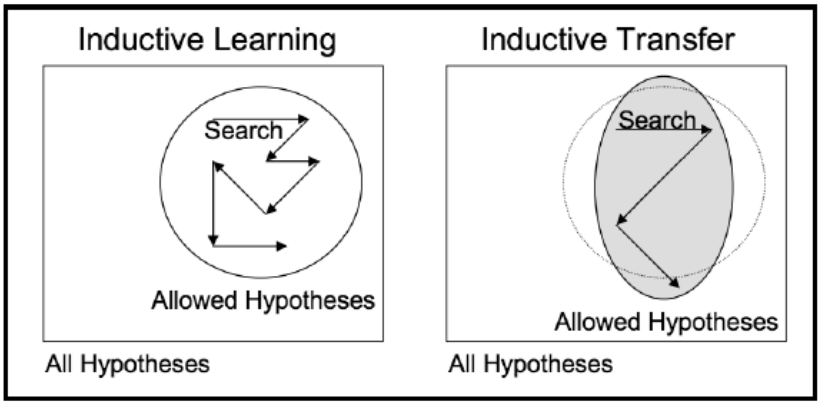


Figure 9. Inductive transfer [30]

In Computer Vision, deep learning has been developed for tasks such as object identification, scenes recognition, using different Convolutional Neural Networks architectures, where lower layers act as feature extractors and the top layers work on the features that are specific to the task [33].

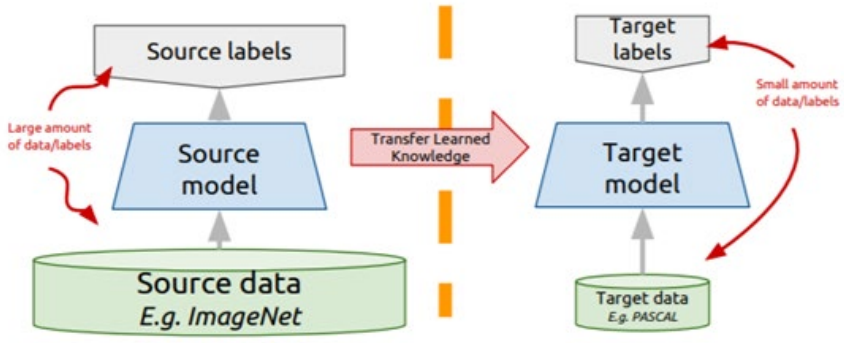


Figure 10. Transfer Learning in Computer Vision[9]

[9] See Appendix 2 for image source.

As computer vision technology advanced, deep learning also progressed in the last years, offering impressive results. State-of-the-art deep learning models have even surpassed the human performance in image recognition.

Because training such an elaborate model requires a lot of resources and time, people use these state-of-the-art models that have had access to that amount of resources to train their model on their specific tasks.

Two main approaches have been preferred when it comes to transfer learning using pre-trained models: (1) pre-trained models as feature extractors, (2) fine-tuning the pre-trained models.

4.3.2 Pre-trained models as feature extractors

Deep learning models learn different features on their architectural layers. These layers are often connected to a final fully connected layer to get the result. This layered architecture enables us to disconnect the final layer from the network and use the rest of the network as a feature extractor.

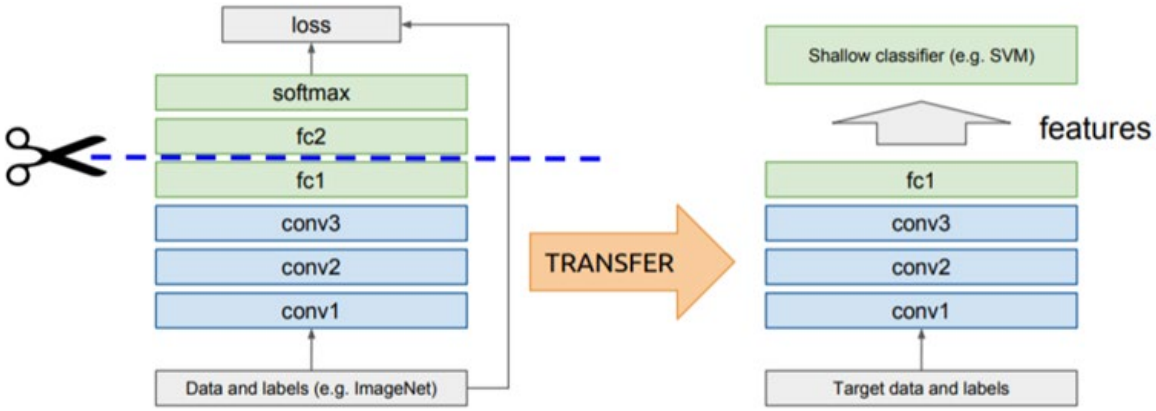


Figure 11. Pre-trained models as feature extractors[10]

Razavian et al. [36] show that these "off-the-shelf CNN" models [35] do perform well to in feature extraction and even out-perform the specialized deep learning models (Figure 12). The experiments have been made on tasks such as object classification, scene classification, plants, animals and buildings recognition etc.

[10] See Appendix 2 for image source.

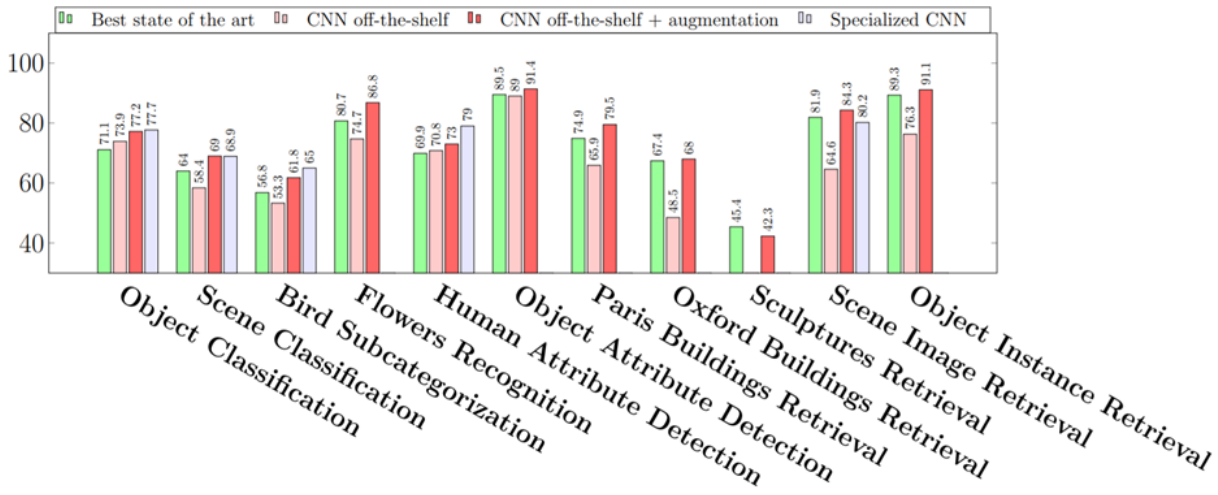


Figure 12. Performance comparison between off-the-shelf pre-trained models and specialized deep learning models [36]

4.3.3 Fine-Tuning Pre-Trained Models

This method does more than disconnecting the output layer and replacing it with a final classification layer, but also involves retraining some of the bottom layers (Figure 13).

In deep learning, bottom layers learn generic features, while the top layers learn more task-specific features, as the architecture is configurable with different hyperparameters. Starting from this, the weights can be fixed on some layers (freeze the layers) while retraining or the layers can be fine-tuned in order to better suit the needs of the task. Therefore, the knowledge of each state is used as a starting point in the retraining of the model. This reduces the time and increases the performance [31].

Freezing a layer means that that the weights of the layer will not be updated during backpropagation. This is an option when the training set labels are minimal, and overfitting becomes a problem. Fine-tuned layers, on the other hand, are updated during backpropagation and is recommended when the training set contains enough data.

Most of the time, different learning rates can be applied on each layer, combining freezing and fine-tuning towards the point of full efficiency (Figure 13) [31].

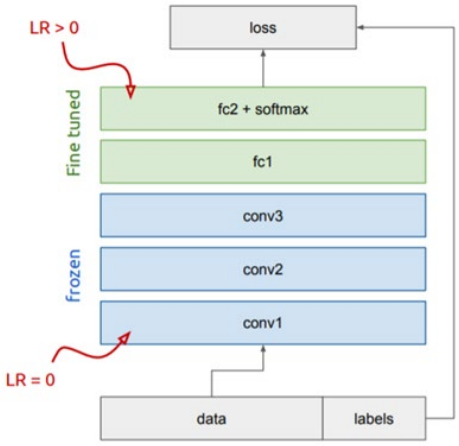


Figure 13. Fine-tuning pre-trained models[11]

Transfer Learning is all about optimizing the process of training a model. There are three possible benefits of transfer learning, as presented by Jude Shavlik [30]: a jump-start given by the knowledge already available at the beginning of the process, a bigger rate of improvement during the training (higher slope), and a higher converging point of the performance rate (higher asymptote).'

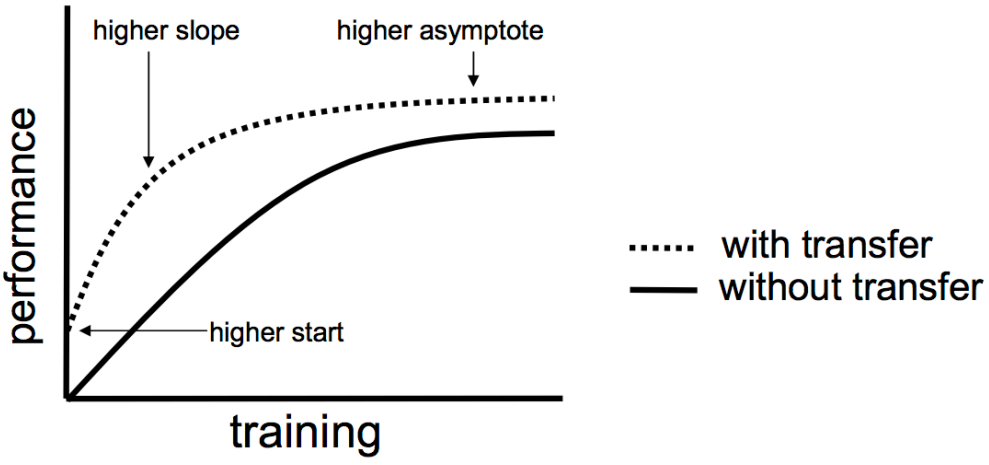


Figure 14. Three benefits of Transfer Learning [30]

[11] See Appendix 2 for image source.

4.4 Pre-Trained Models

The fact that general models are trained on huge and comprehensive datasets, and because training such deep learning models require a great amount of resources, transfer learning is preferred and widely popular in computer vision.

As it was mentioned before, an important step in the successful training of the model is choosing the best machine learning model to apply transfer learning onto. All are state-of-the-art technology, but not all of them suits every project. It is fundamental to have a model that offers good performance on the task it has been trained on.

For computer vision, as well as for other domains of machine learning, a lot of research and training has been done in the previous years. The resulting state-of-the-art deep learning networks are available to the large public and can be acquired freely and easily, both on the web, or directly integrated in machine learning libraries (e.g. Keras).

The most popular and best performing such models are: VGG19, InceptionV3, Xception, ResNet50. These are the ones I considered using in developing the model for this project.

4.4.1 ResNet Architecture

ResNet [37], short for Residual Networks, uses the network-in-network architecture, a collection of building blocks (Figure 15) is comprised in the network itself.

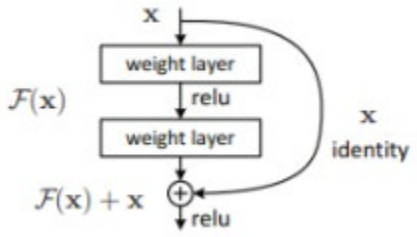


Figure 15. Residual learning: a building block [37]

It allows us to train deeper neural networks with 152 layers. But because of the vanishing gradient problem, ResNet introduced the concept of skip connection (adding the input to the output of the convolutional block). This way, the gradient flows back through these paths. Also, the concept ensures that each layer of the network performs at least as good as the previous layer. The network was further updated in 2016 to use identity mappings in order to obtain higher accuracy [28]. For the architecture visualization and a comparison between ResNet and VGG, see Appendix 1.

ResNet is widely used in building neural networks and training custom models by transfer learning. Pre-trained models become a base model on top of which a final network is added in order to be trained for specific tasks.

ResNet-50 is a smaller version of the original ResNet, consisting of 50 layers of convolutional neural networks. It is built-in the Keras library.

4.4.2 VGG16 Architecture

The VGG16 [22] is a rather simple architecture compared to ResNet. It consists of only 3x3 convolutional layers. Downsizing is made via max pooling, and 2 fully connected layers are followed by a SoftMax classifier (Figure 16).

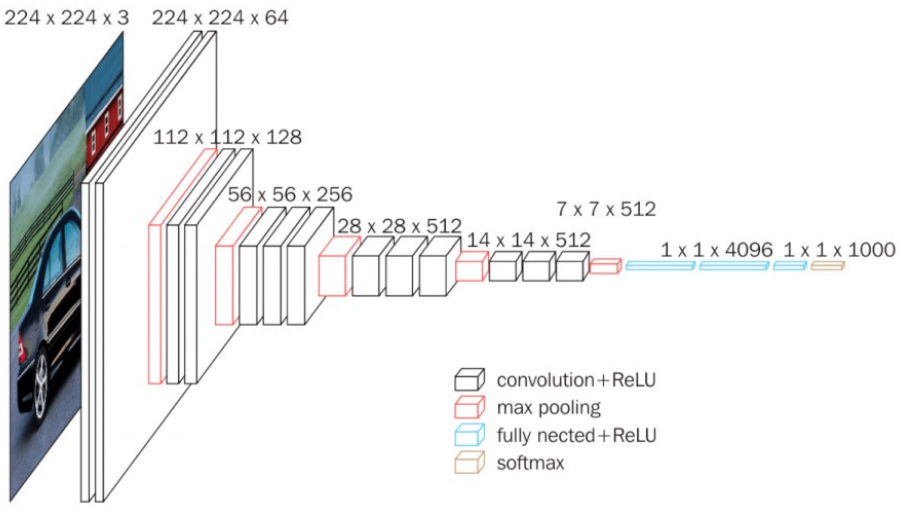


Figure 16. VGG16 architecture[12]

The main disadvantages of this architecture are: (1) it is slow to train, (2) weights size on disk is large.

[12] See Appendix 2 for image source.

5 IMPLEMENTATION

The following chapter contains a step-by-step description of the entire process of building and training the model proposed in Chapter 2.

5.1 Creating the dataset

As presented in Section 4.2, after a period of research, I decided to use the VSD dataset from InterDigital, also filling the blanks of the database with images retrieved from Google.

The process of building the dataset is described in the following sections. The entire project has been coded in Python, using several libraries that will be mentioned during this chapter.

5.1.1 Extracting Frames from the VSD Dataset Videos

As mentioned in Chapter 3, the VSD dataset contains 86 videos downloaded from YouTube, and annotations for the videos.

The database consists of two folders, one for the videos, and one for the annotations. The videos are .MP4 format, while the annotations are stored in .TXT files. The corresponding files from the two folders have the same name. The structure of the database is presented below:

```
YouTube-gen
├── annotations
│   ├── 0egEFZq2Y28.txt
│   ├── 0f-QmF0Avvc.txt
│   ├── 18GLKynpwvM.txt
│   ├── 1KA8kQS2PTE.txt
│   └── (82 more files)
├── videos
│   ├── 0egEFZq2Y28.mp4
│   ├── 0f-QmF0Avvc.mp4
│   ├── 18GLKynpwvM.mp4
│   ├── 1KA8kQS2PTE.mp4
│   └── (82 more files)
```

The annotation files contain a list of integers representing the interval of frames from the corresponding video which are marked as violent scene.

```
591 640
713 762
828 1651
1754 1798
2198 2385
2713 4423
```

To create a dataset for model training, I needed to extract the frames from the videos. To do that, I wrote a Python script that would parse the folders, read the video and text files, and extract the frames according to the frame intervals annotated.

In the process, I used built-in Python libraries, but also the *OpenCV*[13] library. OpenCV provides the means to read video files frame by frame as images and store them into JPEG files.

```
1.  import cv2
2.
3.  cap = cv2.VideoCapture('filename.mp4')
4.  while(cap.isOpened()):
5.      ret, frame = cap.read()
6.
7.      # check for correct interval
8.
9.      cv2.imwrite('filename_frame.jpg', frame)
10. cap.release()
```

The check whether the current frame is within one of the intervals in the annotation files or not results in separating the frames into two categories: (1) violent and (2) non-violent.

At the end of this step, more than 15.000 images depicting violence were extracted. From the non-violent frames, I extracted only approximately 25%, a frame from every second.

5.1.2 Pre-processing the New Dataset

Because the annotation did not contain any labels, manual work of labelling needed to be done. Besides, many of the images resulted from the process of extraction were almost identical, due to being consecutive frames. Therefore, I kept only one of the images that were strikingly similar.

Because the annotations come from video classification, meaning the labelling is made for the entire frames interval, not frame by frame, there were many frames that did not contain a clear and visible action or violent action at all, or the subject was blurred, darkened etc. I removed those from the dataset as well.

After this long and laborious process, the database consisted of only about 1000 images, labelled and organised into folders like this:

```
dataset
├── cold-arms
├── fight
├── fire
├── firearms
├── gore
└── non-violence
```

[13] https://opencv.org/

The actual images have not been processed, I saved them in their original size and shape.

At this point, I also realised that the dataset contains far less images than ideally. So, I started searching on Google Images and download pictures fitting the task. At closer look, the dataset was missing important features, consisting of instances depicting violence or harmful situations, such as gory scenes, mutant creatures, blood and injuries, human and animal bodies, buildings and vegetation on fire. Also, the non-violent set was incomplete. Many counterparts of violent scenes were not present in the non-violent images. I tried to make a complete set by gathering as much general images as possible.

5.2 Dataset Augmentation

Because the dataset is small compared to what a proper dataset would look like, augmentation was helpful to extend my database.

Keras interface allows us to augment the training set after loading the images in memory using *ImageDataGenerator.* The class[14] offers multiple ways to do the augmentation: image rotation, zooming, cropping, horizontal and vertical flipping, range shifting etc.

```
ImageDataGenerator(rotation_range=30,
                   zoom_range=0.15,
                   width_shift_range=0.2,
                   height_shift_range=0.2,
                   shear_range=0.15,
                   horizontal_flip=True,
                   fill_mode="nearest")
```

An example of augmentation using the Keras generator is presented in Figure 17.

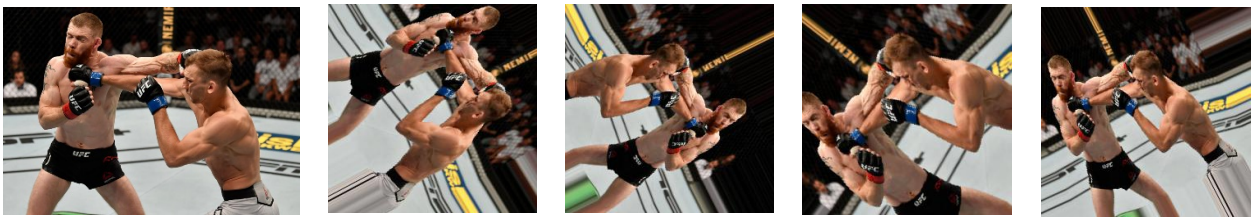


Figure 17. Augmentation example

Another recommended method is *mixup*. This technique was introduced by Zhang et al. in 2017 [38] as a regularization technique. Because we do not know the real distribution of data which can lead to overfitting, mixup comes into help to reduce this problem.

[14] https://keras.io/api/preprocessing/image/#imagedatagenerator-class

Mixup introduces combinations of pairs of images and their labels. A shallow explanation is given by this equation[15], where *t* is the ratio of mixing (a float between 0 and 1). Also, an example of mixing two images is presented in Figure 18.

```
new_image = t * image1 + (1-t) * image2
```

Figure 18. Mixup with a ratio of 30%

5.3 Building the Neural Network

As previously mentioned, for this step I wanted to find the best pre-trained model to use for my model. I tried multiple deep learning models: ResNet50, InceptionV3, Xception, VGG16.

Let us consider the ResNet50 as feature extractor. I disconnect the final layer and build a classifier on top, using the weights from pre-training on the ImageNet dataset.

```
baseModel = ResNet50(weights="imagenet",
                     include_top=False,
                     input_tensor=Input(shape=(IMG_SIZE, IMG_SIZE, 3)))
```

The network I built to classify the violence in images consists of a pooling layer, a few core neural layers and two normalization layers.

```
headModel = baseModel.output
headModel = GlobalAveragePooling2D()(headModel)
headModel = Flatten(name="flatten")(headModel)
headModel = Dense(1024, activation="relu")(headModel)
headModel = Dropout(0.25)(headModel)
headModel = Dense(512, activation="relu")(headModel)
headModel = Dropout(0.5)(headModel)
headModel = Dense(512, activation="relu")(headModel)
headModel = Dense(32, activation="relu")(headModel)
headModel = Dense(6, activation="softmax")(headModel)
```

[15] https://medium.com/ai%C2%B3-theory-practice-business/mixup-a851c9ea14bf

The final layer is a fully connected layer with 6 (as the number of categories) neurons as output. For the *Dense* layers I used ReLU activation function, except for the output layer, where I used SoftMax. After splitting the dataset into training set and validation set of 75%-25%, I used the batches generated by the *ImageDataGenerator* and trained the model for various epochs, 25 to 100. I also use RMSprop, with a learning rate of 10^{-4} as optimizer.

The results started to improve by the time I doubled the size of the database. Also, in the process of training, as I was having problems with overfitting, using the Mixup technique helped me to deal with this issue, increasing the validation accuracy by 3-4% (Table 3).

The best performing deep learning models were the ResNet50 and the VGG16. They both provided similar results, but with variable epochs number (see Table 4). The ResNet model peaks fast, reaching the top validation accuracy after only 13 epochs and maintaining it through the next epochs (up to 100), while other models, such as VGG16, need more training time to do so. In general, the VGG16 required more time for training with the same batch size and dataset size than ResNet50.

Table 3. Mixup impact on training with ResNet50 over 25 epochs

Augmentation	Training Accuracy	Validation Accuracy
w/o mixup	79.89%	65.34%
w/ mixup	72.54%	71.88%

5.4 Wrapping the Predictor in a Script

For the model to be used easily for violence detection in images, I created a script that will allow users to quickly get the predictions of the model on an image. The script will receive as input parameter the path to the image file and will return the resulting category (and other information i.e. the full prediction numbers).

```
> ./detectviolence 'filename.jpg'
Fight [85.54%]

> ./detectviolence -f 'filename.jpg'
Cold weapons    13.46%
Fight           85.54%
Fire            0.02%
Firearms        0.48%
Gore            0.49%
Non-violence    0.01%
----------------------
Fight [85.54%]
```

6 RESULTS

Following the implementation and the test made on several state-of-the-art technologies, I trained a deep learning model that would detect and classify violence in still images with an accuracy of 81%.

As mentioned in Chapter 5, various attempts have been made in order to acquire high performances in terms of accuracy of classification. Figure 19 shows the plot of accuracy and loss values over the training period using ResNet50.

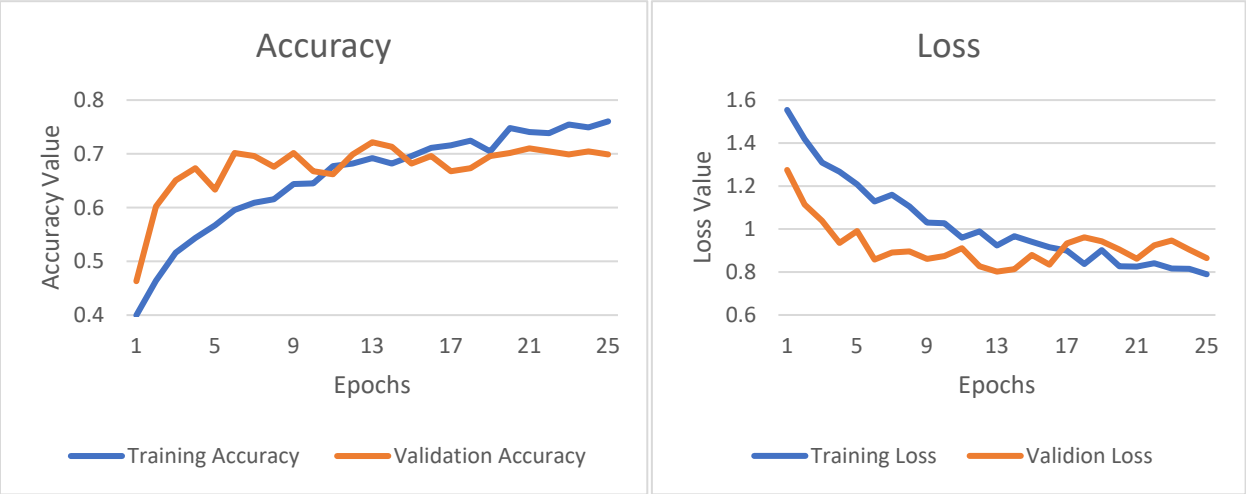


Figure 19. ResNet50 with Image Augmentation performance over 25 epochs with a batch size of 32

The training accuracy rises to 75% after 25 epochs and keeps growing up to 94% when trained without *mixup* (see Table 4), while the validation accuracy begins curve to flatten after 15 epochs to approximately 70%. The loss plot follows an inverted shape as normal, descending from 1.6 to 0.8.

Table 4 shows the evolution of the accuracy rate while training for different epochs on the same database and batch size. A comparison is made between training a model with ResNet50 [24] as feature extractor and training a model with VGG16 [22] as feature extractor, with or without the use of mixup augmentation. It is clear that mixup helps the training process by eliminating the overfitting problem, reducing the difference between the accuracy on the training set and the actual accuracy on validation, especially for VGG16 training. Even though the learning process is slowed down a little by the mixup, in the end the results show that the models have higher accuracy on validation.

Table 4. Comparison of accuracy rate on training for varying epochs (training and validation in %)

	25 Epochs		50 Epochs		75 Epochs		100 Epochs	
ResNet50 w/o mixup	79.89%	65.34%	90.56%	71.59%	91.63%	73.86%	94.01%	70.85%
VGG16 w/o mixup	70.31%	73.58%	79.69%	77.56%	84.94%	81.25%	89.23%	80.68%
ResNet50 w/ mixup	72.54%	71.88%	82.58%	65.62%	86.36%	66.76%	88.83%	71.02%
VGG16 w/ mixup	64.11%	69.89%	73.39%	76.99%	78.50%	80.97%	83.52%	80.68%

In Table 5, we can see a comparison between the performances of the models that have been trained with different feature extractors (see section 5.3). As mentioned several times before, it was necessary to choose the pretrained model that will suit this project well, and to do it from the beginning was crucial. I did quick trainings on these models and I could observe right from the start that some of them (i.e. InceptionV3 [23], Xception) will not going to perform well, while the others showed a good potential. The final accuracy rates are presented below:

Table 5. Performance of model trained with various feature extractors

Pretrained Model	Accuracy
InceptionV3	30%
Xception	41%
ResNet50	74%
VGG16	**81%**

The prediction performance is also measured by precision and recall [39]. Judging these values, we can observe that some categories are better classified than others. We have *firearms* with good precision on one hand, and *cold weapons* on the other hand, which has the lowest score. These differences in values are generated by the fact that some categories are harder to judge (as debated in Chapter 1).

Finally, the model I built using the steps described in Chapter 5 and based on the VGG16 pre-trained model is the one that performed the best, with an accuracy of 81%, as shown in Table 5. Table 6 shows the precision, recall and f1-score.

Table 6. Precision, recall and f1-score of the final model

Category	Precision	Recall	F1-score
Cold weapons	0.67	0.69	0.68
Fight	0.78	0.86	0.82
Fire	0.93	0.82	0.87
Firearms	0.90	0.62	0.73
Gore	0.82	0.79	0.80
Non-violence	0.79	0.89	0.84

These scores can be better illustrated by some examples of classification. The following results are obtained by feeding images to the model:

Gore (99.56%)

Fight (97.95%)

Fire (48.23%)

Figure 20. Correct predictions[16]

Figure 21 shows that the model has room for improvements. The following images does not fall under the right category, the predictions made by the model are wrong. The first image is mistakenly classified as a gory image mainly because the model learned to classify people on fire as gore, therefore, the firefighter being close to the fire, the image is labeled as gore. The next image is problematic due to its bad resolution (observable in the full-size image), the darkness around the subjects and also due to the weird pattern on the trousers, that may look similar to an open-flesh wound. The last image is classified as depicting a fight with a 47.22% rate, although the next-best classification is 'non-violence (43.27%)'.

[16] See Appendix 2 for image sources.

Gore (83.73%)

Gore (61.95%)

Fight (47.22%)

Figure 21. Incorrect predictions[17]

The main problem of violence detection, as presented earlier in this paper, is the relativity of violence understanding. Many times, two different persons will label the same image differently because their vision on violence is different. Figure 22 shows examples of such images that people can argue on their correct classification.

Non-violence (77.01%)

Non-violence (90.75%)

Non-violence (84.56%)

Figure 22. Debatable predictions[17]

The first image shows a body under a sheet; however, this is only indicated by the dark vignette and by the title of the original article. In other circumstances, a child might not see a dead body in the picture. The second image shows a man that separate two other men that are fighting. The scene does not contain enough features for the model to classify it as a fight scene, although the context clearly suggests that. In the third image it might be unclear whether the children are playing, hugging, or fighting. The model tends to be lenient on this kind of images, labeling the debatable ones as non-violent.

[17] See Appendix 2 for image sources.

7 CONCLUSIONS AND FUTURE WORK

In this project, I developed a deep learning model that recognizes violent scenes that would have an emotional impact over an 8-year old child by using deep neural networks. I built the model by aggregating the knowledge of a pre-trained model and a classification network, with VGG16 being the appropriate state-of-the-art model for the task. The model reports whether a violent or harmful scene is depicted in the image and outputs the class predicted and the score.

The outcome of this project shows that there is a lot to be done in this area of technology. Children can be protected using state-of-the-art computer vision technology and, by building a model that would detect the images that can be harmful to see for them and that classify the violence depicted in still images, I believe people can be encouraged to address this issue more. Building machine learning models and using them in all kinds of applications will, eventually, make the world safer for children.

During the work of this project, I learnt a lot about Deep Learning, Convolutional Neural Networks, Computer Vision, Machine Learning in general. I started from the bottom, knowing only a few things about this domain, but guided by my advising professor I managed to get into the machine learning science. On this journey, I understood that much of the technology has already been implemented, and that machine learning is available to anyone. I was surprised to see how easily one can train a model that would do such elaborate work. Thanks to people who invested their time and knowledge, we can now build powerful AI models with a few lines of codes.

I also encountered problems, such as lack of datasets biased towards different tasks, and I learnt that most of the hard work is focused on gathering all the resources you need rather than training the models itself.

The motivation for starting the project came from the specifics of the job I had during the last 2 years. Being part of the developing team for a parental control application made me aware of the dangers present on the Internet. We decided that adding my work to this application would be helpful for parents.

There are many aspects that can be improved. Future work will strive to increase the accuracy of the model. This can be acquired by gathering more data and by tuning the model better. Each class has its unique features and there is work to be done to refine the database of each violence category and to identify the features that will increase the accuracy of prediction. As the model will improve, it can be integrated in the parental application that will allow live detection of violence in accessed images.

The dataset used, the code for the implementation and the results are available on GitHub [40].

REFERENCES

1. Browne, Kevin & Hamilton-Giachritsis, Catherine. (2005). *The influence of violent media on children and adolescents: A public-health approach*. Lancet. p. 8
2. National Scientific Council on the Developing Child. (2010). *Persistent Fear and Anxiety Can Affect Young Children's Learning and Development: Working Paper No.* 9. Retrieved from www.developingchild.harvard.edu.
3. Guerra, Nancy & Knox, Lyndee. *Classification, The Causes of Violence, Prevention and Control of Violence, Bibliography.* Retrieved 2020-05-31, from https://law.jrank.org/.
4. Krug et al., *World report on violence and health*. Archived 2015-08-22 at the Wayback Machine, World Health Organization, 2002.
5. *Community Standards*, Facebook, Inc. Retrieved 2020-05-31, from www.facebook.com/communitystandards.
6. Potter, M.C., Wyble, B., Hagmann, C.E. et al. (2014). *Detecting meaning in RSVP at 13 ms per picture*. Atten Percept Psychophys 76, 270–279. https://doi.org/10.3758/s13414-013-0605-z
7. Schmidhuber, Jürgen (2015). *"Deep Learning"*. Scholarpedia.
8. Cireșan, D.; Meier, U.; Schmidhuber, J. (2012). *Multi-column deep neural networks for image classification*. 2012 IEEE Conference on Computer Vision and Pattern Recognition.
9. Krizhevsky, Alex; Sutskever, Ilya; Hinton, Geoffrey (2012). *ImageNet Classification with Deep Convolutional Neural Networks.* NIPS 2012: *Neural Information Processing Systems*, Lake Tahoe, Nevada.
10. "*Google's AlphaGo AI wins three-match series against the world's best Go player*". TechCrunch. 25 May 2017.
11. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei. (2009). *ImageNet: A Large-Scale Hierarchical Image Database.* IEEE Computer Vision and Pattern Recognition (CVPR).
12. Bojarski, M., Firner, B. et al. (August 17, 2016). *End-to-End Deep Learning for Self-Driving Cars*. Retrieved 2020-06-05, from https://devblogs.nvidia.com/deep-learning-self-driving-cars/.
13. Esteva, A., Kuprel, B., Novoa, R. et al. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature* 542, 115–118. https://doi.org/10.1038/nature21056
14. C.H. Demarty, C. Penet, M. Soleymani, G. Gravier. (2014). *VSD, a public dataset for the detection of violent scenes in movies: design, annotation, analysis and evaluation*. In Multimedia Tools and Applications, May 2014.
15. C.H. Demarty, B. Ionescu, Y.G. Jiang, and C. Penet. (2014). *Benchmarking Violent Scenes Detection in movies*. In Proceedings of the 2014 12th International Workshop on Content-Based Multimedia Indexing (CBMI), 2014.

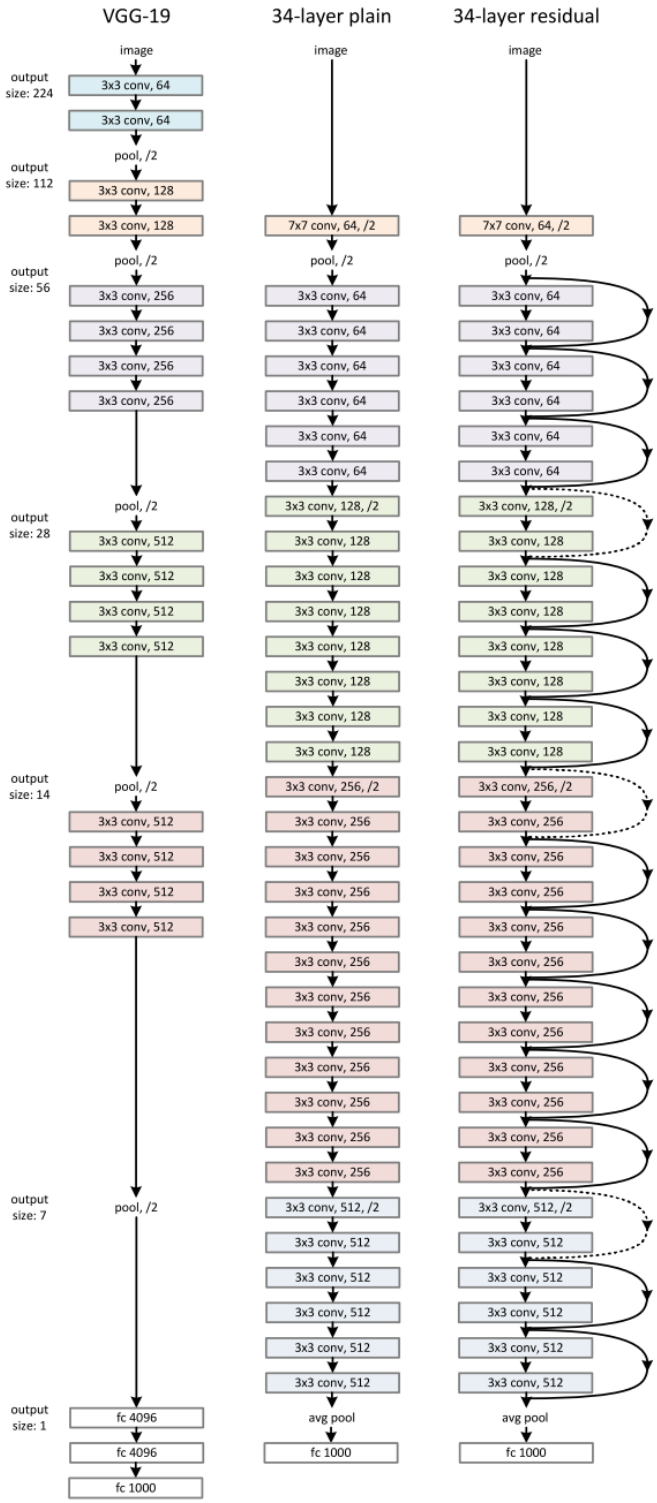
16. M. Sjöberg, B. Ionescu, Y.G. Jiang, V.L. Quang, M. Schedl and C.H. Demarty. (2014). *The MediaEval 2014 Affect Task: Violent Scenes Detection*. In Working Notes Proceedings of the MediaEval 2014 Workshop, Barcelona, Spain (2014)
17. C.H. Demarty, C. Penet, G. Gravier and M. Soleymani. (2012). *A benchmarking campaign for the multimodal detection of violent scenes in movies*. *In Proceedings of the 12th international conference on Computer Vision* – Volume Part III (ECCV'12), Andrea Fusiello, Vittorio Murino, and Rita Cucchiara (Eds), Col. Part III. Springer Verlag, Berlin.
18. Olmos, R., Tabik, S., & Herrera, F. (2018). *Automatic handgun detection alarm in videos using deep learning*. Neurocomputing, 275, 66-72.
19. Ming C., Kunjing C. and Ming L. (2019). "*RWF-2000: An Open Large-Scale Video Database for Violence Detection*." arXiv preprint arXiv:1911.05913.
20. O. Deniz, I. Serrano, G. Bueno and T. Kim, "Fast violence detection in video," 2014 International Conference on Computer Vision Theory and Applications (VISAPP), Lisbon, 2014, pp. 478-485.
21. Zhou P, Ding Q, Luo H, Hou X. (2018). *Violence detection in surveillance video using low-level features*. PLoS ONE 13(10): e0203668.
22. Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. CoRR, abs/1409.1556.
23. Szegedy, C., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., & Anguelov, D. et al. (2015). Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr.2015.7298594
24. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr.2016.90
25. Metz, Cade (November 9, 2015). "Google Just Open Sourced TensorFlow, Its Artificial Intelligence Engine". *Wired*. Retrieved 2020-06-08, from https://www.wired.com/2015/11/google-open-sources-its-artificial-intelligence-engine/.
26. Keras | TensorFlow Core. TensorFlow. (2020). Retrieved 2020-06-08, from https://www.tensorflow.org/guide/keras.
27. Dwivedi, P. (2019). *Understanding and Coding a ResNet in Keras*. Towards Data Science. Retrieved 2020-06-08, from https://towardsdatascience.com/understanding-and-coding-a-resnet-in-keras-446d7ff84d33.
28. He K., Zhang X., Ren S., Sun J. (2016) Identity Mappings in Deep Residual Networks. In: Leibe B., Matas J., Sebe N., Welling M. (eds) Computer Vision – ECCV 2016. ECCV 2016. Lecture Notes in Computer Science, vol 9908. Springer, Cham
29. Brownlee, J. (December 20, 2017). *A Gentle Introduction to Transfer Learning for Deep Learning*. Machine Learning Mastery, in *Deep Learning for Computer Vision*. Retrieved

2020-06-09, from https://machinelearningmastery.com/transfer-learning-for-deep-learning/.

30. Soria Olivas, E. et al. (2010). *Handbook of research on machine learning applications and trends*. Information Science Reference.
31. Sarkar, D. (November 15, 2018) *A Comprehensive Hands-on Guide to Transfer Learning with Real-World Applications in Deep Learning*. Towards Data Science. Retrieved 2020-06-10, from https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a.
32. Wang, T., Huan, J., & Zhu, M. (2019). Instance-Based Deep Transfer Learning. *2019 IEEE Winter Conference on Applications of Computer Vision* (WACV). https://doi.org/10.1109/wacv.2019.00045
33. Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014*). How transferable are features in deep neural networks?* ArXiv, abs/1411.1792.
34. Mutegeki, R., & Han, D. (2019). Feature-Representation Transfer Learning for Human Activity Recognition. *2019 International Conference on Information and Communication Technology Convergence (ICTC).* https://doi.org/10.1109/ictc46691.2019.8939979
35. Shin et al. (2016). Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. arXiv, abs/1602.03409
36. Razavian, A.S., Azizpour, H., Sullivan, J., Carlsson, S. (2014). CNN Features off-the-shelf: An Astounding Baseline for Recognition. arXiv, abs/1403.6382
37. He, K., Zhang, X., Ren, S., Sun, J. (2015). *Deep Residual Learning for Image Recognition.* arXiv: abs/1512.03385
38. Zhang, H., Cisse, M., Dauphin, Y., & Lopez-Paz, D. (2017). *mixup: Beyond Empirical Risk Minimization*. arXiv: abs/1710.09412
39. Olson, D.L., Delen, D. (2008) *Advanced Data Mining Technique.* Springer, 1st edition (February 1, 2008), p. 138.
40. Grigore, E.M. (2020). *Graphic Content Detection in Images and Classification of Types of Violence Using Deep Neural Networks*. Available: https://github.com/edwmrkg/ml-violence
41. Rosebrock, A. (March 20, 2017). *ImageNet: VGGNet, ResNet, Inception, and Xception with Keras*. Retrieved 16 June 2020, from https://www.pyimagesearch.com/2017/03/20/imagenet-vggnet-resnet-inception-xception-keras/.
42. Foote, Keith D. (February 7, 2017). A Brief History of Deep Learning. Retrieved 2020-06-05 from https://www.dataversity.net/brief-history-deep-learning/.

APPENDIX 1: RESNET ARCHITECTURE VS. VGG ARCHITECTURE

The following image represents the VGG-19 architecture on the left and the ResNet architecture on the right [37].



APPENDIX 2: IMAGE SOURCES

Figure 1. Firearms classification

- https://www.dw.com/en/why-german-police-officers-rarely-reach-for-their-guns/a-17884779, last accessed on 12-06-2020.
- https://www.alaskapublic.org/2019/04/02/police-man-shot-killed-after-pointing-bb-gun-at-officers/, last accessed on 12-06-2020.
- https://c8.alamy.com/comp/R6B6DX/dangerous-criminal-man-with-gun-stealing-car-of-scared-young-woman-on-outdoor-parking-R6B6DX.jpg, last accessed on 12-06-2020.

Figure 2. Cold weapons classification

- https://www.darksword-armory.com/wp-content/uploads/2018/07/the-vindaaris-sword-fantasy-medieval-weapon-1328-darksword-armory.jpg, last accessed on 12-06-2020.
- https://i.ytimg.com/vi/gydQjwP8Li0/maxresdefault.jpg, last accessed on 12-06-2020.
- https://www.123rf.com/photo_673234_gangster-with-a-knife-threatening-young-woman.html, last accessed on 12-06-2020.

Figure 3. Fire classification

- https://upload.wikimedia.org/wikipedia/commons/7/79/Operation_Upshot-Knothole_-_Badger_001.jpg, last accessed on 12-06-2020.
- https://www.aljazeera.com/mritems/imagecache/mbdxxlarge/mritems/Images/2020/1/26/, last accessed on 12-06-2020.
- https://www.mirror.co.uk/news/uk-news/flames-of-fury-tibetan-man-turns-772693, last accessed on 12-06-2020.

Figure 4. Fight scenes classification

- https://www.martialtribes.com/how-to-develop-explosive-kick-power/, last accessed on 12-06-2020.
- https://www.pinterest.com/pin/247768416982592673/, last accessed on 12-06-2020.
- https://www.ancient.eu/article/1010/battle-of-teutoburg-forest/, last accessed on 12-06-2020.

Figure 5. Gore classification

- https://www.telegraph.co.uk/football/2017/10/20/dynamo-kiev-defender-domagoj-vida-suffers-gruesome-head-injury/, last accessed on 12-06-2020.
- https://www.dappercadaver.com/products/decapitated-victim-dummy, last accessed on 12-06-2020.
- https://bloodandgutsforgrownups.wordpress.com/2011/09/03/10-incredibly-terrifying-horror-movie-monsters/, last accessed on 12-06-2020.

Figure 6. Non-violent images classification

- https://www.nbc15.com/content/news/Oshkosh-police-officer-released-from-hospital-after-stabbing-incident-566003251.html, last accessed on 12-06-2020.
- https://www.theverge.com/2020/2/7/21126966/coronavirus-handshake-ban-mwc-disease-cold-flu, last accessed on 12-06-2020.
- https://www.publicdomainpictures.net/ro/view-image.php?image=211575&picture=campfire, last accessed on 12-06-2020.
- https://www.jumpropedudes.com/workouts/male-fitness-model, last accessed on 12-06-2020.

- https://www.akc.org/expert-advice/lifestyle/quiz-are-you-a-dog-person-or-a-cat-person/, last accessed on 12-06-2020.
- https://www.gffoodservice.com.au/idea/cooking-terms-and-definitions/, last accessed on 12-06-2020.

Figure 7. Steps to creating the model
- https://i.stack.imgur.com/K9lmg.png

Figure 8. Traditional learning vs Transfer learning
Figure 10. Transfer Learning in Computer Vision
Figure 11. Pre-trained models as feature extractors
Figure 13. Fine-tuning pre-trained models
- https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a, last accessed on 15-06-2020.

Figure 16. VGG16 architecture
- https://neurohive.io/en/popular-networks/vgg16/, last accessed on 17-06-2020.

Figure 20. Correct predictions
- https://www.abc.net.au/news/2019-11-19/queensland-bushfires-ten-children-allegedly-start-fires/11717444, last accessed on 22-06-2020.
- https://cali-adventures.tumblr.com/post/30055189829/halloweenmakeup-halloween-gross-gruesome/amp, last accessed on 22-06-2020.
- https://www.dailymail.co.uk/news/article-3380402/Bare-knuckle-boxing-ritual-Peru-men-women-fight-one-settle-disputes-avenge-loved-ones.html, last accessed on 22-06-2020.

Figure 21. Incorrect predictions
- https://www.cbc.ca/natureofthings/episodes/into-the-fire, last accessed on 22-06-2020.
- https://www.parentmap.com/calendar/playgarden-community-campfire, last accessed on 22-06-2020.

Figure 22. Debatable predictions
- https://dailypost.ng/2014/10/03/peace-maker-dies-separating-fight-landlord-tenant/, last accessed on 22-06-2020.
- https://parenting.firstcry.com/articles/dealing-with-fighting-between-twin-children/, last accessed on 22-06-2020.
- https://www.welovelotto.com/news/lottery-winners-neighbours-bankrupt/, last accessed on 22-06-2020.