Rúbia Reis Guerra

Deep Learning for Accessibility: Detection and Segmentation of Regions of Interest for Sign Language Recognition Systems

Belo Horizonte - Minas Gerais

November, 2019

Rúbia Reis Guerra

Deep Learning for Accessibility: Detection and Segmentation of Regions of Interest for Sign Language Recognition Systems

Final thesis presented to the Undergraduate Program in Systems Engineering of the Federal University of Minas Gerais in partial fulfillment of the requirements for the degree of Bachelor of Science in Systems Engineering.

Federal University of Minas Gerais - UFMG

Electrical Engineering Department - DEE

Machine Intelligence and Data Science Laboratory - MINDS

Supervisor: Frederico Gadelha Guimarães

Co-supervisor: Tamires Martins Rezende

Belo Horizonte - Minas Gerais

November, 2019

Acknowledgements

To my research, internship, and thesis supervisor, Prof. Frederico Gadelha, for his kind, extensive guidance.

To Prof. Gisele Pappa, for patiently introducing me to the world of research.

To MSc. Tamires Rezende and to Giulia Zanon, for guiding me along this thesis and for everything they have taught me about the world of sign language recognition.

To my family and friends, for supporting me every step of the way. A special thanks to my friends Artur Mello, Ítalo Andrade, and Maísa Cota, for their willingness to listen and for reminding me that friendship goes beyond the classroom.

To Hương, for being my daily company and my closest friend throughout my undergraduate journey, and for proofreading this thesis.

To my colleagues at MINDS Lab and e-Speed, for our shared learning experiences and for our shared beers.

To Prof. Ana Liddy and Júlio Carvalho, for their unwavering cordiality and promptness to help every Systems Engineering student.

To my System and Electrical Engineering professors, for guiding me through the world of Engineering.

To my internship colleagues at AppProva, Enacom, and Tenbu, for contributing to my personal and professional growth.

And finally, to Universidade Federal de Minas Gerais and all its employees, to CNPq and to every citizen that has helped fund my higher education.

"Dear Lord, let the equations be linear, the noise be Gaussian, and the variables be separable." (Terrance J. Sejnowski)

Resumo

Mais de 6% da população mundial apresenta perda auditiva incapacitante. Para a comunidade surda, a comunicação é um desafio diário que precisa ser superado, uma vez que as línguas de sinais são consideravelmente menos prevalentes do que as línguas faladas. Nesse cenário, a tradução de sistemas que efetivamente permitem a comunicação entre surdos e ouvintes tem o potencial de melhorar o acesso a serviços básicos, como saúde e educação, para milhões de pessoas em todo o mundo.

Progresso considerável tem sido feito em soluções que permitem a compreensão básica de informações textuais por pessoas surdas. Até recentemente, no entanto, a natureza visual complexa das linguagens de sinais e a escassez de conjuntos de dados disponíveis para treinar soluções baseadas em aprendizado de máquina causaram um descompasso no avanço da tecnologia tradução automática de sinais.

Neste contexto, a ascensão das técnicas de Aprendizado Profunda (Deep Learning) é promissora para preencher a lacuna entre as soluções de reconhecimento de sinais e de texto-para-sinal. Este estudo tem como objetivo apresentar os principais conceitos sobre a teoria de Deep Learning e como ela pode ser aplicada para contribuir no projeto de um sistema de reconhecimento de linguagem de sinais robusto. Em particular, apresenta-se a aplicação da arquitetura Mask R-CNN para realizar a segmentação semântica de vídeos de sinais de Língua Brasileira de Sinais, capaz de identificar as regiões de interesse com uma taxa de detecção de 66.3%.

Palavras-chave: Deep Learning, Reconhecimento Automático de Sinais, Segmentação Semântica, Mask R-CNN.

Abstract

Over 6% of the world's population presents disabling hearing loss. For the deaf community, communication is a daily challenge that has to be overcome, since sign languages are considerably less prevalent than spoken languages. In this scenario, translating systems that effectively enable communication between the deaf and the hearing to have potential to increase access to basic services, such as healthcare and education, to millions of people around the world.

Considerable progress has been made in solutions that allow for basic comprehension of textual information by deaf people. Until recently, however, the complex visual nature of sign languages and the paucity of datasets available to train machine learning-based solutions caused a mismatch in the advancement of automatic sign translation technology.

In this context, the rise of Deep Learning techniques seems promising to bridge the gap between sign recognition and text-to-sign solutions. This study aims to introduce the main concepts regarding Deep Learning theory and how it can be applied to contribute to the design of a robust sign language recognition system. In particular, this study presents an application of Mask R-CNN to perform semantic segmentation in Brazilian Sign Language videos, which is capable of identifying regions of interest with a detection rate of 66.3%.

Keywords: Deep Learning, Sign Language Recognition, Semantic Segmentation, Mask R-CNN.

List of Figures

List of abbreviations and acronyms

2D	Two-dimensional
3D	Three-dimensional
ADALINE	Adaptive Linear Neuron
AE	Autoencoder
ASL	American Sign Language
CNN	Convolutional Neural Networks
CNPq	Conselho Nacional de Desenvolvimento Científico e Tecnológico
COCO	Common Objects in Context
CPM	Convolutional Pose Machines
DBN	Deep Belief Networks
DEE	Departamento de Engenharia Elétrica
DL	Deep Learning
FC	Fully Connected
HMM	Hidden Markov Models
HPE	Human Pose Estimation
ILSVRC	ImageNet Large Scale Visual Recognition Competition
IOU	Intersection Over Union
LIBRAS	Brazilian Sign Language
LSTM	Long Short-Term Memory
mAP	Mean Average Precision
MINDS	Machine Intelligence and Data Science

MLP	Multilayer Perceptron
PASCAL VOC	Pattern Analysis, Statistica Modelling and Computational Learning Visual Object Classes
R-CNN	Regions with CNN
RBM	Restricted Boltzmann Machine
RNN	Recurrent Neural Networks
RPN	Region Proposal Networks
ROI	Region of Interest
SPAI	Secure and Private AI
SS	Semantic Segmentation
SVM	Support Vector Machines
TL	Transfer Learning
UFMG	Universidade Federal de Minas Gerais
VOC2012	PASCAL Visual Object Classes 2012
WHO	World Health Organization
WSHP	Weakly and Semi-Supervised Human Body Part Parsing

Contents

Chapter 1

Introduction

According to the World Health Organization (WHO), 6.1% of the world's population suffers from disabling hearing loss. While 6.1% may seem a small percentage, it amounts to 466 million people around the world, of which 10 million are located in Brazil [WHO, 2019]. Yet, the numbers are even more staggering once the ease of communication is taken into account. For instance, a study by Kuenburg et al. [2016] highlights the gaps in global health knowledge between the deaf and hearing communities, pointing to communication as a major barrier for health care access and understanding of common medical terminology.

Communication is a two-way, continuous process. If either party involved cannot decode each other's message, then the process is incomplete and therefore ineffective. Deaf people's primary form of communication is sign language: a composition of manual signaling, such as palm orientation and hand configuration, and non-manual cues, such as facial expression and body positioning. Sign languages are generally unique within each culture, with their own grammar and lexicon [Sandler and Lillo-Martin, 2006]. Moreover, it is also worth noting that most deaf children and adults has poorer literacy than their hearing peers despite having the same level of cognitive capacities [Mehravari et al., 2017].

In this context a question emerges, how can technology bridge the gap between the deaf and hearing communities? The answer may lie in advancements of studies on bi-directional sign language translation. A tool enabling the translation of spoken and written messages to sign language can be helpful in establishing a one-way route of communication. Similarly, a system capable of interpreting signs, translating them to written and spoken languages, completes the cycle of communication. While applications involving translations from text to signs present a greater level of development, as shown by HandTalk [HandTalk, 2019] and Suíte VLibras [VLibras, 2019], automatic sign language recognition (SLR) solutions still require improvement before achieving the capacity of enabling natural conversations between deaf and hearing individuals [Cheok et al., 2019].

According to Er-Rady et al. [2017], there are three main challenges to developing a reliable sign language recognition system:

1. *Visual complexity*: sign languages are fully visual and involve multiple parameters at the same time - hands, face and body - while the majority of the meaning of a sign is carried through the hands. Although a slight change in one of the hands' configurations can represent a completely different or an undefined sign, it is almost impossible for a human signer to repeat the same sign with the exact same hand locations and trajectories. Moreover, hand/hand and hand/face occlusions may happen, hiding information that could be crucial to help distinguish among signs.

2. *Scarcity of databases*: because different recordings of a sign can present great variability due to the nature of movement being executed, an extensive annotated database is required to allow for the design and validation of a robust sign recognition system. However, due to the cost and the workload involved in the creation of such structured data sets, there are still very few of them available. One solution explored by researchers in the field is to create smaller data sets, including a limited number of signs recorded at highly controlled environments. While this workaround enables the exploration of new techniques for sign language recognition, little can be affirmed concerning whether or not the proposed solutions can be generalized to the language as a whole. Moreover, solutions designed in a particular language, e.g., American Sign Language (ASL), may not be applicable to other languages, such as Brazilian Sign Language (Libras).

3. *Underdeveloped linguistics*: as a result of the previous issue and in combination with a lack of active researchers interested in the field, few sign languages have been formally documented. Also, because a universal sign language does not exist, researchers' efforts generally are restricted to structuring their homeland's language.

While the second and third issues require resources and knowledge that are beyond the scope of this project, the problem of processing complex visual information can be tackled through Computer Vision and Machine Learning approaches. The next section briefly discusses the state of automatic sign language recognition systems, highlighting how the two areas of knowledge can contribute to the design of robust recognition systems.

1.1 Sign Language Recognition Systems

The general framework for sign language recognition is comprised of three fundamental parts. The first consists of an annotated database representative of the language. As discussed before, there are few sign language data sets available, and most of them contain only a small amount of signs from the researchers' national sign language, recorded under controlled conditions.

The second block consists of the definition of procedures for extracting significant characteristics, or "features", from a sign's recording. These features generally revolve around posture and trajectory of the hands, since most of a sign's information is contained in the manual parameters. The choice of feature extraction is decisive to the final performance of the system. One common way of addressing the visual complexity issue in the feature extraction stage is through the use of *wearables* sensors [Kawamoto et al., 2018], which enable tracking of coordinates of selected points through video frames or cameras with *time-of-flight* technology [Almeida et al., 2014, Escobedo-Cardenas and Camara-Chavez, 2015, Filho et al., 2017] which is capable of capturing depth along with RGB channels.

The last stage of a system for sign recognition consists of a Machine Learning algorithm capable of learning to recognize a sign using the features extracted in the previous stage. Many different techniques have been explored, varying from ensemble methods, Hidden Markov Models (HMM) and Support Vector Machines (SVM) [Er-Rady et al., 2017].

In spite of achieving high precision on classifying signs, approaches that utilize external artifacts in the second stage are not applicable to everyday encounters, where, ideally, the sign translation system would be mostly used. Accordingly, with the development of more advanced computational techniques, such as Convolutional Neural Networks (CNNs) Krizhevsky et al. [2012], external artifacts are being replaced by a combination of Machine Learning and Computer Vision based approaches [Rawat and Wang, 2017, Zhang et al., 2017].

For this purpose, this thesis explores the application of Deep Learning models to perform segmentation of regions of interest (ROI) for a general sign language recognition system. The base architecture is shown on Figure 1, and was proposed as an extension of the work presented in Zanon et al. [2019]. The scope of this thesis is highlighted in the figure.

Deep Learning allows for the discovery of intricate structures underlying large amounts of data and have dramatically advanced knowledge in image and video processing [LeCun et al., 2015] thus posing as an interesting solution to eliminate dependency on any kind of external equipment. The Venn diagram in Figure 2 shows the relation among different levels of Artificial Intelligence (AI) technology, positioning Deep Learning as a subset of Machine Learning.

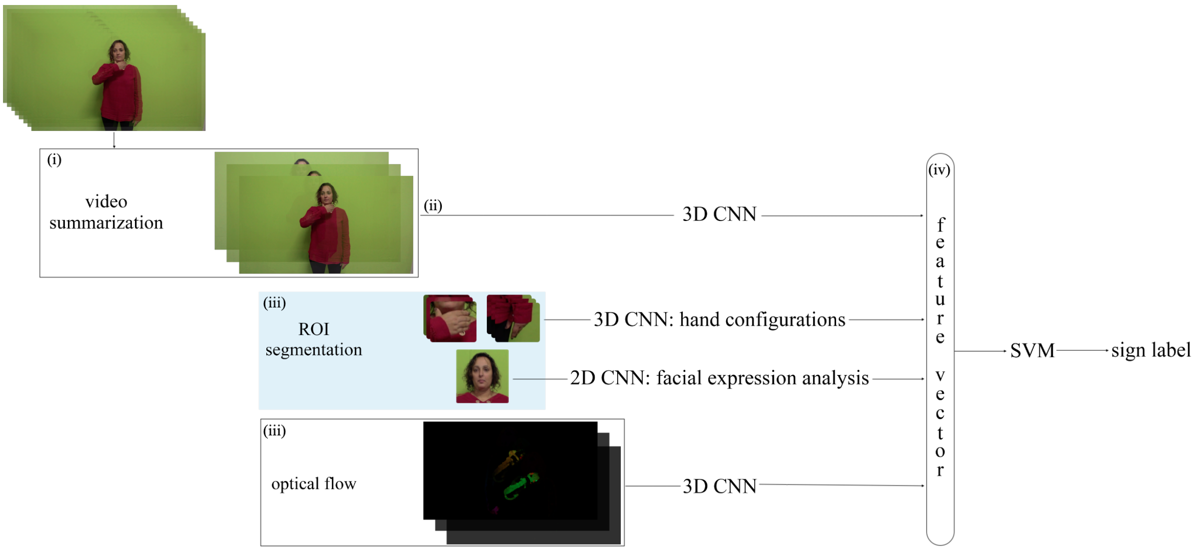


Figure 1 – Proposed architecture of a sign language recognition system. (i) The original video is summarized in key frames. (ii) The selected frames feed a 3D Convolutional Neural Network (CNN). (iii) Segmented regions of interest (ROI) and the optical flow are also extracted from the summarized video. (iv) Finally, all the information is transformed into a feature vector, which is the input to a one-versus-all multi-class Support Vector Machine (SVM) to classify the sign.

Source: the author

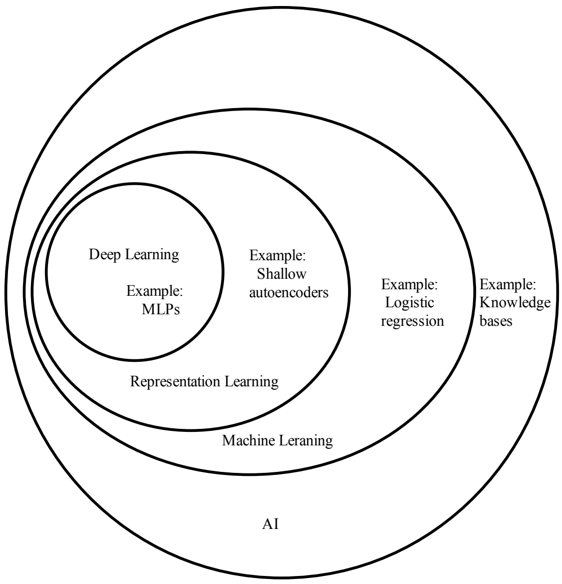


Figure 2 – Venn diagram showing how Deep Learning relates to general Artificial Intelligence. Deep Learning is a part of Machine Learning, which in turn is a subset, and not the entirety, of AI technology.

Source: Goodfellow et al. [2016]

1.2 Objectives

The main objective of this senior thesis project is to propose a robust ROI extraction technique for sign language recognition systems. The following activities also encompass the scope of this thesis:

- review of state-of-the-art neural networks architectures applied to video data;
- review of state-of-the-art techniques for semantic segmentation and pose estimation, discussing how each technique can contribute to mitigate the effects of visual complexity in SLR;
- investigation of the effects of Machine Learning systems in society;
- implementation and test of a Deep Learning-based ROI extractor.

Previous studies regarding sign language recognition have exhaustively explored the social relevancy of such systems [Almeida, 2014, Rezende, 2016, Almeida, 2017, Mendes de Assis, 2018]. Hence, this work explores, instead, the historical evolution of Deep Learning. Particular attention is given to social impacts resulting from recent improvements in Machine Learning techniques. Also, while the end goal of this project is to propose a methodology for ROI extraction in sign language recognition systems, it is worth noting that the recognition of signs is beyond the scope of this project.

1.3 Outline

The remainder of the work is organized as follows: Chapter 2 introduces the main concepts related to Deep Learning, while Chapter 3 discusses the evolution of Deep Learning from historical and social standpoints. Chapter 4 introduces a brief survey on semantic segmentation and pose estimation. Chapter 5 summarizes the methodology utilized in this project and discusses the experimental results. Chapter 6 presents the final observations.

Chapter 2

Deep Learning

Data representation can vastly impact the performance of a Machine Learning algorithm [Najafabadi et al., 2015]. To illustrate, suppose there is a simple learning method capable of separating different categories in a 2D space by using only a straight line. For each of the representations of the dataset shown in Figure 3, such method would perform significantly better if the input is in the form of polar coordinates.

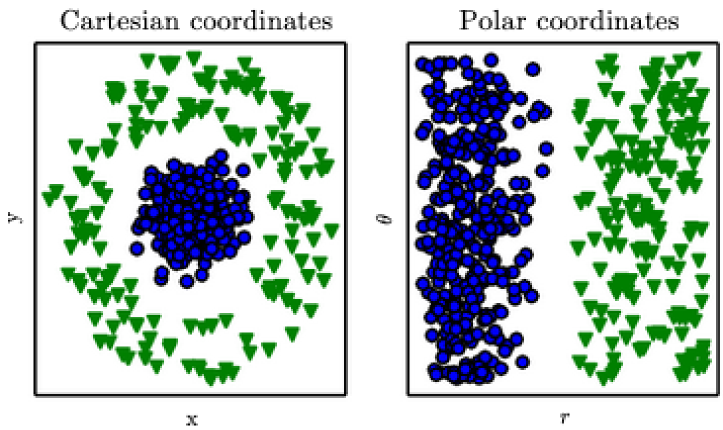


Figure 3 – Example of different representations of a dataset

Source: Goodfellow et al. [2016]

It can be very difficult, however, to extract high-level, abstract features from raw data. The passage below captures the main issues around data representation:

> A major source of difficulty in many real-world artificial intelligence applications is that many factors of variation influence every single piece of data we are able to observe. The individual pixels in an image of a red car might be very close to black at night. The shape of the car's silhouette depends on the viewing angle. Most applications require us to disentangle the factors of variation and discard the ones that we do not care about [Goodfellow et al., 2016].

Traditional data engineering, at first glace, does not seem to be effective in obtaining a representation of the problem when it is nearly as complicated as solving the original problem. Deep Learning solves this central dilemma by introducing representations that are expressed in terms of other simpler representations. In short, Deep Learning enables the computer to build complex concepts out of simpler concepts [Goodfellow et al., 2016].

This chapter is organized as follows: Section 2.1 introduces the main concepts around Deep Learning methods. Section 2.2 expounds each of the three main layers of Convolutional Neural Networks. Finally, the chapter ends in discussing the concept of Transfer Learning, a technique that has been widely used to improve the performance of training deep neural networks, especially in Computer Vision domain.

2.1 Deep Learning Architectures

Before diving into the different types of Deep Learning (DL) architectures and their applications, it is important to define the common terminology underlying DL theory. First, the concept of "learning" in DL actually comprises of three different processes through which a model can be learned [Lison, 2012]. This is also applicable to Machine Learning methods in general:

1. Supervised learning: in this modality, training data comes in observation-label pairs. The end goal is to derive a model that generalizes well to new data, i.e., by being capable of mapping new, unlabelled observations in the label space;

2. Unsupervised learning: in cases where only a collection of inputs is available, it is possible instead to derive underlying patterns, i.e. describe possible correlations between features, cluster observations in a few groups based on similar behavior and detect *outliers*;

3. Reinforcement learning: the basic components of these types of systems involve perceptions, actions, and rewards. A *agent* interacts with a *environment* and is rewarded depending on whether or not its actions are in accordance with the main purpose of the system. This way, the goal of the agent is to learn the behavior that maximizes its expected cumulative reward over time.

The second group of concepts to be briefly discussed in this thesis relate to Artificial Neural Networks (ANN) theory. In ANN, a *perceptron* is a learning algorithm that maps input[1]-output relations through a set of weights and a step function. While the perceptron model has many limitations due to its linear nature, multilayer perceptrons

[1]In case of 2D data, for example, inputs are (x_{1i}, x_{2i}) pairs for each i-th observation

(MLP) can handle many nonlinear problems. MLPs can combine multiple layers of artificial *neurons* units. As opposed to the original perceptron, each layer can contain a different activation function depending on its specific purpose in the network [Nielsen, 2015]. An artificial neuron can be represented by the diagram of Figure 4.

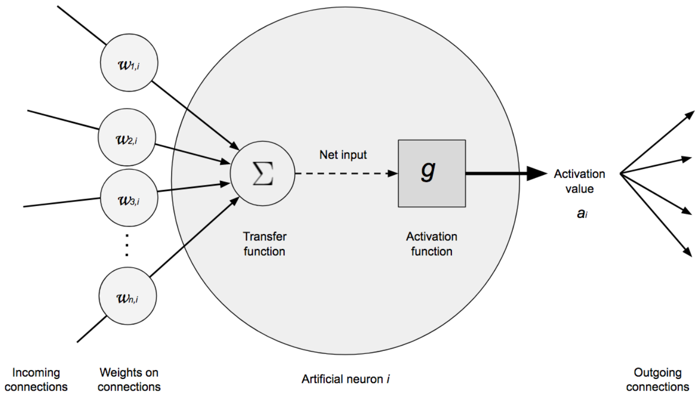


Figure 4 – Artificial neuron in a multilayer percetron neural network

Source: Patterson and Gibson [2017]

The last concept corresponds to the difference between *generative* and *discriminative* models [Patterson and Gibson, 2017]. Generative models try to understand *how* the data was created by learning the joint probability distribution $p(x, y)$. Discriminative models, on the other hand, focus on the *conditional* probability distribution $p(y|x)$, i.e., given an input x, this class of models tries to discriminate to which output y the input can be mapped to.

Deep Learning architectures can be categorized with respect to the basic method they are derived from, the type of learning employed, whether the models employed are generative or discriminative and with regard to the topology of the network employed [Guo, 2017]. Figure 5 summarizes the main methods, presented along with a non-exhaustive list of examples of architectures.

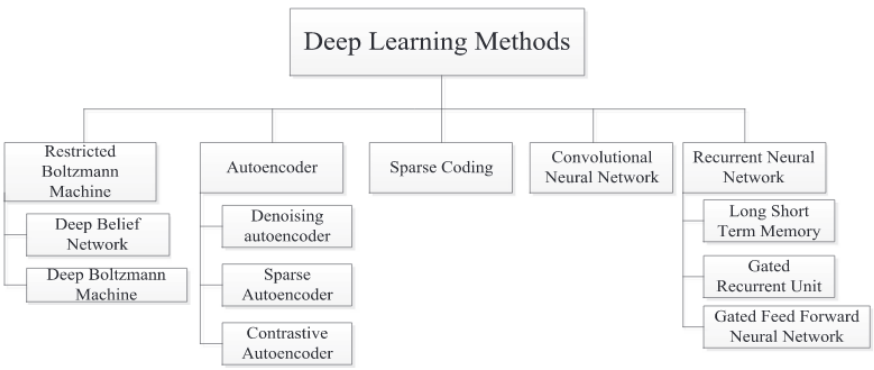


Figure 5 – General taxonomy for Deep Learning architectures

Source: Guo et al. [2018]

To illustrate some of the methods shown in Figure 5, the remainder of this section briefly touches Autoencoder (AE) and Recurrent Neural Networks (RNN). Section 2.2 introduces in greater detail Convolutional Neural Networks, a class of DL architectures widely applied to Computer Vision problems.

Autoencoder

Autoencoders (AE) are generally used to reduce the dimensionality of a dataset [Hinton and Salakhutdinov, 2006]. The structure of AE networks is, at a first glance, highly similar to that of an MLP, as Figure 6 shows. One of the key differences in relation to MLPs, however, is that AE architectures present the same number of units in the input and output layers. The hidden layers in the AE of Figure 6 also exemplifies the intuition behind its compression capabilities, since the input must pass through a bottleneck before being expanded back to the output layer.

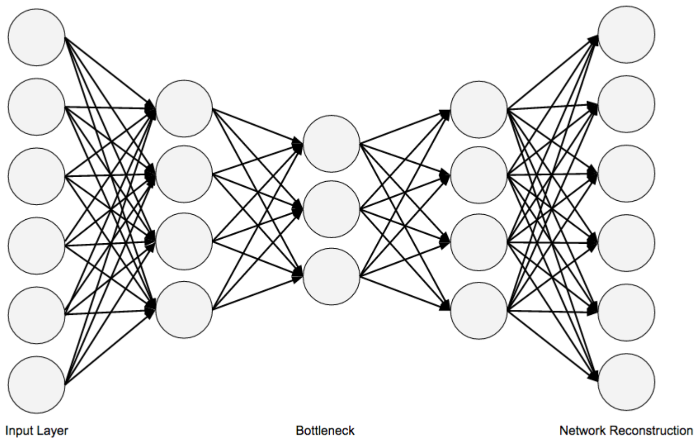


Figure 6 – Example of a compression autoencoder

Source: Patterson and Gibson [2017]

Another aspect that separates AE from MLP models is that the first perform unsupervised learning of unlabeled data. For instance, a *denoising* AE that is trained over multiple variations of "corrupted" data, e.g., in which features are removed randomly, can learn to identify the "uncorrupted" output. By learning to understand the difference between the input and output representations instead of focusing on the output itself, AEs serve as boosters in anomaly detection systems [Patterson and Gibson, 2017].

Recurrent Neural Networks

Recurrent Neural Networks (RNN) were developed to handle sequential data. Juergen Schmidhuber, a lead researcher in Deep Learning, provides an interesting explanation on RNNs:

> [Recurrent Neural Networks] allow for both parallel and sequential computation, and in principle can compute anything a traditional computer can compute. Unlike traditional computers, however, Recurrent Neural Networks are similar to the human brain, which is a large feedback network of connected neurons that somehow can learn to translate a lifelong sensory input stream into a sequence of useful motor outputs. The brain is a remarkable role model as it can solve many problems current machines cannot yet solve. [Patterson and Gibson, 2017]

One of the most important characteristics of this class of methods is that it allows for dynamic changes of its elements. The behaviour of hidden layers, in this case, might be determined not only by the activations in the previous layers, but also by activations at earlier times [Nielsen, 2015]. This effect makes RNNs particularly interesting candidates to problems where data presents some form of time-dependency, such as in time series [Che et al., 2018] and speech processing [Ahmed et al., 2018]. Another promising application of RNNs is in video understanding tasks, such as in sign and gesture recognition [Wang et al., 2018b], since hybrid CNN-RNN architectures allow for capturing of both visual and temporal information. The next section explores in depth the main components of CNNs.

2.2 Convolutional Neural Networks

The main advantage that Convolutional Neural Networks offer in relation to the other concepts discussed in this chapter is that the architecture takes advantage of spatial relationships present in the data [Guo et al., 2018]. The idealization of CNNs was motivated by minimal data preprocessing requirements and rely on the ideas of *local receptive fields*, *shared weights* and *pooling* [Nielsen, 2015]. Each of these ideas are combined to form the general structure of CNNs, seen on Figure 7.

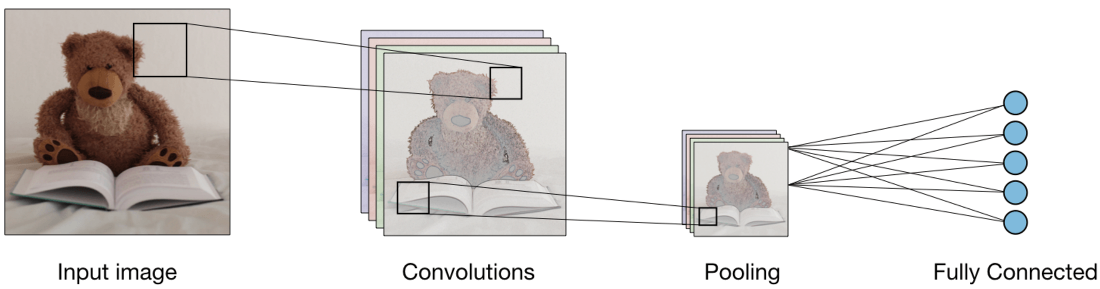


Figure 7 – General architecture of a Convolutional Neural Network

Source: [Amidi and Amidi, 2018]

The layers shown in Figure 7 will be discussed in more detail below. These layers be tweaked, combined, and interleaved, enabling the creation of a multitude of different frameworks. Figure 8 compares a few of the most popular CNN architectures found in literature.

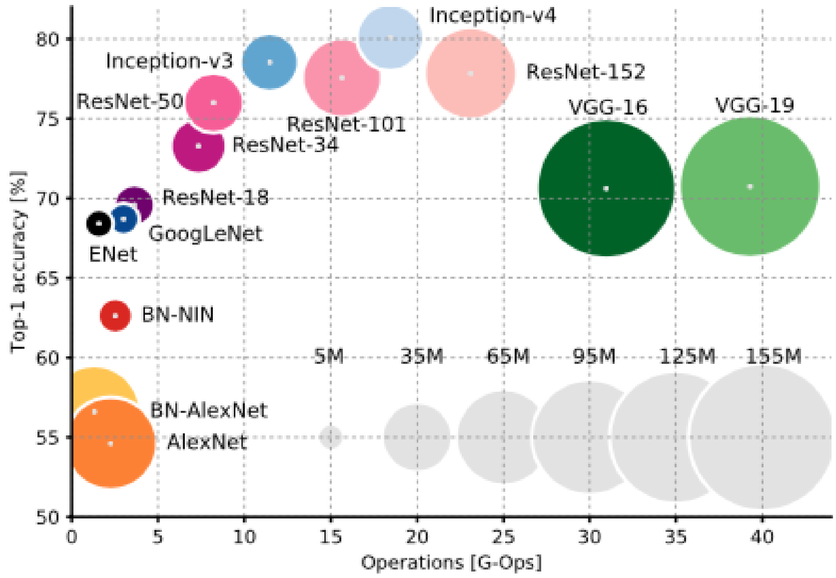


Figure 8 – Comparison of accuracy and computational cost across multiple CNN architectures trained on ImageNet data. Top-1 one-crop accuracy versus amount of operations required for a single forward pass. The size of the blobs is proportional to the number of network parameters; a legend is reported in the bottom right corner, spanning from 5×10^6 to 155×10^6 parameters.

Source: Canziani et al. [2016]

Besides using MLP-like networks in the last layer and being comprised of "units" that function in a similar way as artificial neurons, the intuition of a CNN as a Neural Network can also be associated to its two-stage, supervised learning process. In the first stage, the *forward pass*, each layer feeds the following with *activations*, found by associating a set of weighted inputs and bias through an activation function. In the second stage, the *backward pass*, a loss cost is calculated over the predicted and real responses. The gradient of each parameter of the network with respect to the loss cost is then propagated *backwards* through the hidden layers, updating the weights that connect a layer to the next one. This method of updating the weights is know as *backpropagation* [Guo, 2017].

2.2.1 Convolutional Layer

In the first hidden layer, a local receptive field corresponds to the sub-region of the input image to which a hidden unit[2] is connected to. Likewise, each unit of the following layers is only connected to a limited number of units in the previous layer. Local receptive fields forming $n \times n$ patches move over a layer with fixed step sizes, referred to as *stride*. A stride of length 1 was used in the example shown in Figure 9.

[2] A "unit" in CNNs is analogous to a "neuron" in MLPs

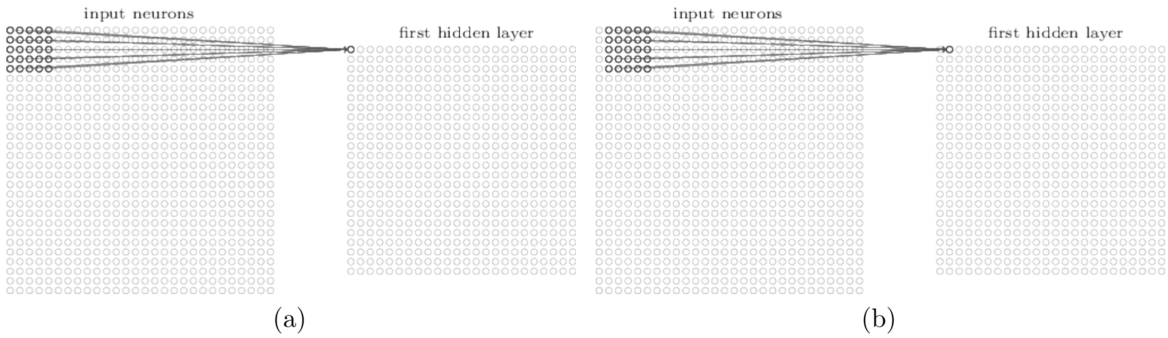


Figure 9 – Examples of local receptive fields in a CNN. There is one hidden unit in the first hidden layer for each local receptive field. A unitary stride was used to move the local receptive field seen in 9a, forming a new input region to the next unit, as seen in 9b.

Source: Nielsen [2015]

In Figure 9, like in the MLP model, each connection from the input layer to a unit in the next layer has a weight. Moreover, each unit has an activation function (σ), that operates over the weighted activations of the previous layer according to Equation 2.1:

$$\sigma\left(b + \sum_{l=0}^{n}\sum_{m=0}^{n} w_{l,m} a_{j+l,k+m}\right) \tag{2.1}$$

in which b is the bias, n is the size of local receptive field, a is the activation of the unit in the previous layer and w is the weight that connects the previous layer's unit to the next layer's one. The activation function can be linear or non-linear, depending on the purpose of the layer. The name *convolutional* comes from the fact that the operation described in Eq. 2.1 is also known as *convolution* [Nielsen, 2015].

Contrary to the MLP model each of the 24×24 hidden units shown in Figure 9 share the same weights and same bias, forming a *feature map*. Consequently, all units in the first hidden layer of Figure 9 detect the same kind of input pattern. This is beneficial since it greatly reduces the number of parameters involved in a CNN.

As a general case, a same hidden layer is constituted of multiple feature maps, also called *filters* or *kernels*. While filters are spatially smaller than the input image, they are more in-depth. Therefore, if an image is composed of three channels, e.g., RGB channels, the filter's height and width will be spatially smaller, but the depth extends up to all three channels.

Examples of outputs of convolutional layers is shown in Figure 10. Each image corresponds to a feature map. Although it is not intuitive to describe what each feature detector is learning, the existence of a spatial structure among the feature maps is clear. At the end of all convolution layers, it is expected that the CNN will have learned features that allow a clear separation of all classes that the network was trained to recognize [LeCun et al., 2015].

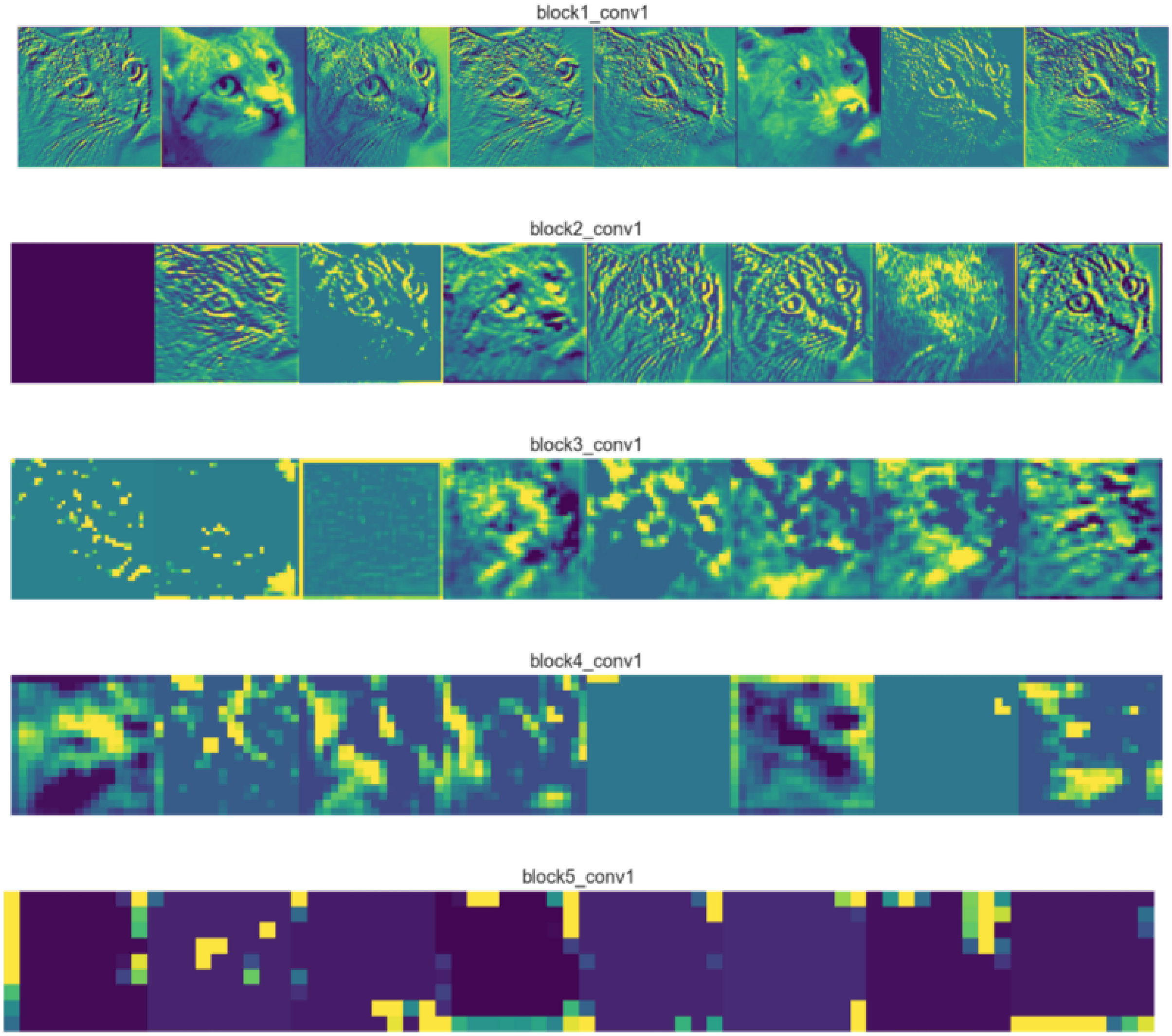


Figure 10 – Feature maps learned in the convolutional stage

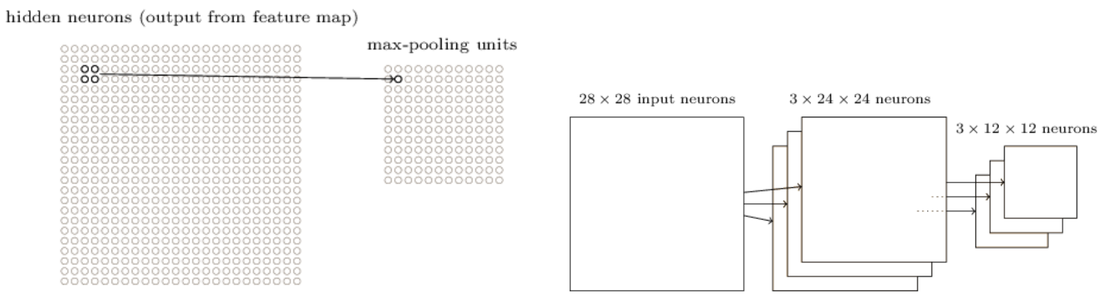
Source: [Dertat, 2017]

2.2.2 Pooling Layer

Pooling layers are generally inserted in-between convolutional layers, simplifying the information outputted from a convolutional stage. By decreasing the dimensionality of the representation and, thus, reducing the number of parameters in the network, pooling layers control overfitting and reduce the computational cost throughout the network. Furthermore, pooling also permits the extraction of dominant features, which are rotational and positional invariant [Nielsen, 2015].

The most common form of pooling operation is the *max-pooling*, which consists of

$n \times n$ filters[3] applied to the activation of the hidden neurons' output from the previous layer. The pooling unit simply performs a maximum operation in the input region, as illustrated by Figure 11:



(a) (b)

Figure 11 – Example of a max-pooling operation. 11a Max-pooling operation applied to a 24×24 output from a convolutional layer, resulting in a 12×12 activation map after pooling is carried. 11b Reduction of three 24×24 activation maps to 12×12 after pooling.

Source: Nielsen [2015]

Apart from max-pooling, L2[4] and average pooling are also widely used. In L2 pooling, the procedure follows as described before, but instead of applying a maximum operation, the square root of the sum of squares of each activation in the filter region is mapped to the next layer. The same intuition applies to average pooling, where the average of the activations in the filtered region is carried to the next layer of the network. In short, although these three types of pooling are most generally applied, there are many choices of operations that can be adapted to boost the network's performance [Guo, 2017].

2.2.3 Classification Layer

The network pipeline ends in one or more fully-connected (FC) layers that perform classification based on the features extracted by the previous layers. The term "fully-connected" comes from the fact that every node of a layer is connected to every node of the next layer. Typically, the classification layer correspond to a set of traditional MLPs, ending with a *softmax* activation function that outputs the probabilities predicted for each class [Nielsen, 2015].

Since most of the CNN's parameters concentrate in the FC layers, training can be computationally expensive. Recently, however, a wide variety of visual recognition tasks have incorporated transfer learning approaches, preserving or fine-tuning parameters pre-trained on the ImageNet dataset [Russakovsky et al., 2015] and adapting the final

[3] $n = 2$ is most generally used
[4] L2 pooling uses a L2-norm operator

FC layers to the problem in hand, as a solution to boost training efficiency [Guo, 2017]. Figure 12 shows a schematic example of transfer learning using pre-trained weights from convolutional layers.

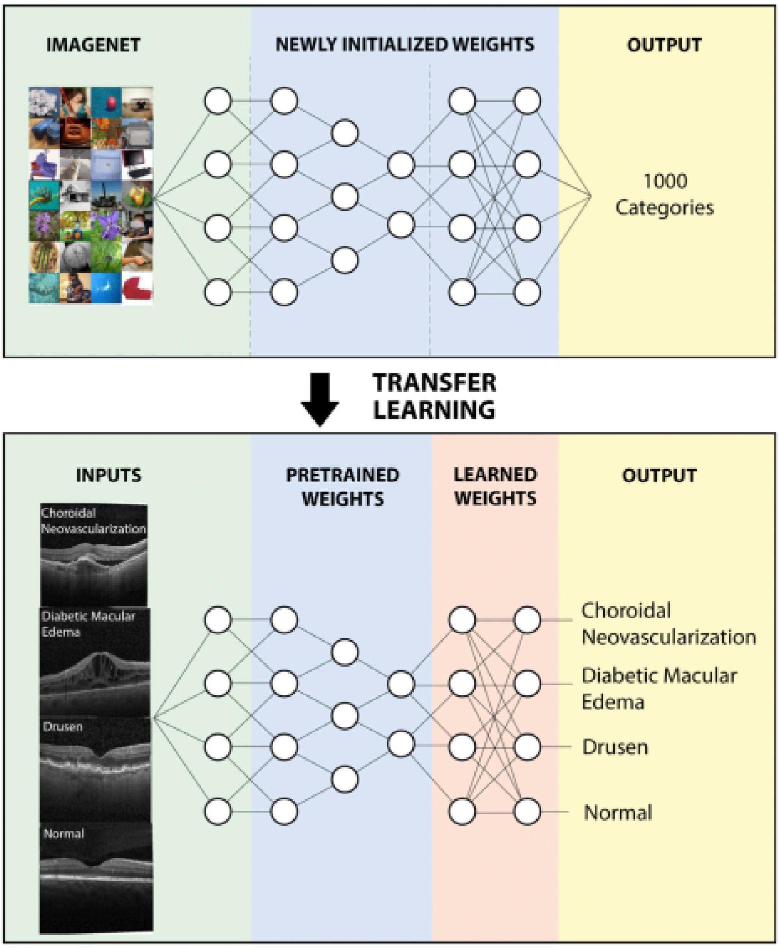


Figure 12 – Example of transfer learning for analysis of retinal OCT images

Source: Zia et al. [2018]

2.3 Transfer Learning

Transfer learning (TL) is used to improve a learner in one domain by transferring information from a related domain. To illustrate, a person with an extensive music background will be able to learn an instrument in a more efficient manner by transferring previously learned musical knowledge to the task of learning to play a new instrument [Weiss et al., 2016]. The same way a person is able to take information from a previously learned task and use it to complement learning of a related task, transfer learning can be applied to Deep Learning, leveraging the specialization obtained on pre-trained models to reduce the computational cost of fully training deep networks and the need for a large amounts of labeled data.

Research on transfer learning aims to build learning machines that can generalize across different domains following different probability distributions [Long et al., 2017]. The main challenges of transfer learning revolve around how to reduce the shifts in data distributions across domains. Meanwhile, multiple studies in a wide variety of applications have reported benefits from adopting TL [Huh et al., 2016].

In Computer Vision problems, deriving a general-purpose CNN to perform classification on ImageNet data and then fine-tuning for a new target task has become a common practice, even in fields that are seemingly unrelated to the domain in which the network was pre-trained [Huh et al., 2016]. Lee et al. [2019], for instance, reported an increase in both accuracy and training speed while applying TL from ImageNet to the development of a fully connected network for a vision-based steel slab identification system. Likewise, Hu et al. [2018] and Yaguchi and Nixon [2018] demonstrated promising results in incorporating transfer learning to perform instance segmentation[5] in a wide variety of scenes. In effect, transfer learning can be applied in the process of training a CNN to perform human segmentation in sign language videos. Since few annotated sign language datasets are available, a transfer learning based approach fills this gap by leveraging large, public datasets, such as ImageNet or COCO [Lin et al., 2014].

[5]Instance segmentation is the task of predicting a mask for each object in an image

Chapter 3

Social Impacts of Deep Learning Research

The epigraph of this thesis is a personal account found on Terrence J. Sejnowski's book The Deep Learning Revolution [Sejnowski, 2018]. Professor Sejnowski is one of the pioneers of Neural Networks research [Ackley et al., 1985], along many other grand minds such as Geoffrey Hinton and John Hopfield [Hopfield, 1982]. The whole passage reads:

> As a graduate student in the Physics Department at Princeton, I approached the problem of understanding the brain by writing down equations for networks of nonlinearly interacting neurons and by analyzing them, much as physicists have over the centuries used mathematics to understand the nature of gravity, light, electricity, magnetism, and nuclear forces. Every night before bed, I would pray: "Dear Lord, let the equations be linear, the noise be Gaussian, and the variables be separable." These are the conditions that lead to analytic solutions, but because neural network equations turn out to be nonlinear, the noise associated with them non-Gaussian, and the variables nonseparable, they do not have explicit solutions. Moreover, simulating the equations on computers at that time was impossibly slow for large networks; even more discouraging, I had no idea whether I had the right equations. [Sejnowski, 2018]

This personal account introduces a few of the reasons why Neural Networks theory, despite having its starts in the 1940s-1960s [McCullough and Pitts, 1943, Hebb, 1949, Rosenblatt, 1958], undergone multiple changes of fortune before achieving its current popularity.

The history behind Deep Learning is long and rich. In effect, capturing all the nuances of its past is a herculean task. This chapter attempts to summarize the key trends that led to recent developments in the area. The chapter ends with considerations regarding some of the current social concerns around the development of AI systems.

3.1 Historical Remarks

What is known today by "Deep Learning" has gone through several different names: *cybernetics* in the 1940s–1960s, *connectionism* in the 1980s–1990s, and the current resurgence under the name Deep Learning beginning in 2006 [Goodfellow et al., 2016]. Each of these phases not only reflected different perspectives, but also marked the popularity of the field in the succeeding years. Figure 13 summarizes the milestones of Deep Learning development.

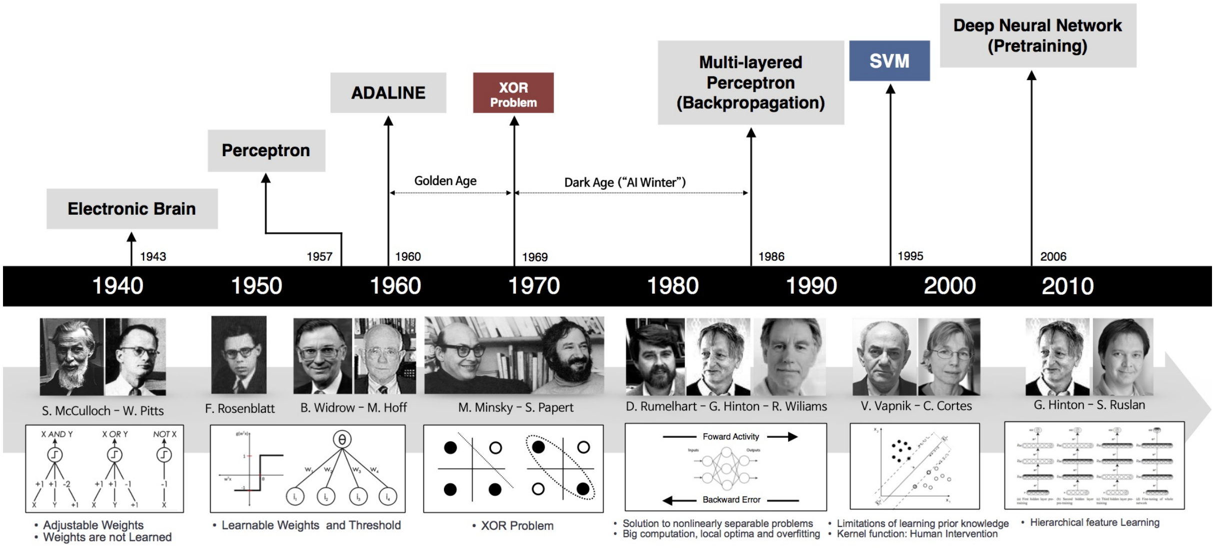


Figure 13 – Timeline of the development of Neural Networks

Source: Beam [2017]

Cybernetics: 1940s–1960s

The first wave of Neural Networks theory was tied to a transdisciplinary approach known as Cybernetics [Wiener, 1948]. Cybernetics connected multiple areas from Control Systems and Electrical Network Theory to Evolutionary Biology and Neuroscience.

McCulloch-Pitts neuron [McCullough and Pitts, 1943] marks the beginning of Neural Networks research, an early mathematical model of how the brain functions. McCulloch-Pitts model mimics the thought process through a "threshold logic", which could recognize two different categories by associating a set of weighted input values $w_1x_1, \ldots, w_nx_n$ to an output y by testing if a function $f(x, w) = \sum_{i=1}^{n} w_ix_i$ is positive or negative. Hebb [1949] took the idea of threshold logic even further, contributing towards quantifying brain processes. The Hebbian Synaptic Plasticity proposed that neural pathways strengthen with successive usage, in particular between neurons that tend to fire at the same time.

A great computational limitation of the McCulloch-Pitt's model was that the weights w_i had to be set by the human operator for each different task. Rosenblatt's (1958) Perceptron circumvented this problem, becoming the first model that could learn a set of weights by analyzing input observations from each category. Shortly after, Widrow and Hoff's (1960) Adaptive Linear Neuron (ADALINE) could also learn to predict real numbers from data.

Another important biological contribution of this period is Hubel and Wiesel's (1962) report on properties of single neurons, recorded with a micro-electrode. Although the impacts of this research for Neural Networks models were not immediately apparent, it inspired the creation of Deep Learning architectures, which are organized in a similar fashion as the hierarchy of areas in the visual cortex studied by Hubel and Wiesel.

Using models with polynomial activation functions and statistical analysis, Alexey Ivakhnenko and Valentin Lapa displayed embryonic efforts towards developing algorithms similar to today's Deep Learning approaches [Ivakhnenko and Lapa, 1967, Ivakhnenko, 1971]. Through a laborious, manual process, the best features were statistically chosen and forwarded on to the next layer of computations.

In 1969, after Minsky and Papert's (1969) book "Perceptron", Neural Networks research entered its first winter. Critics of Rosenblatt's Perceptron pointed the limitations of working with a linear model, most famously, by describing its inability to represent a XOR function. The book provoked great backlash against biologically inspired models, causing Neural Networks research to remain nearly dormant for a decade.

Connectionism: 1980s–1990s

Parallel Models of Associative Memory, a workshop organized by Geoffrey Hinton and James Anderson in 1979, brought together many Neural Networks pioneers [Sejnowski, 2018]. This second wave of Neural Networks researchers were influenced by Cognitive Science, an interdisciplinary approach to understanding thought, learning, and mental organization [Goodfellow et al., 2016]. While cognitive scientists in early 1980s studied symbolic models of reasoning, that were difficult to understand in terms of how they occur in the brain through neurons, connectionists models' were grounded by the actual biology of the brain [Touretzky and Hinton, 1985], even reviving some of Donald Hebb's ideas [Hebb, 1949].

In connectionism, the central idea is that a large number of simple computing units configured as a network can achieve intelligent behavior. This insight is just as applicable to neurons in biological nervous systems as to hidden units computational models. For instance, according to Hinton et al.'s (1986) concept of distributed representation, a system's input should be represented by multiple features. Such features, in turn, should be capable to represent as many inputs as possible. The excerpt bellow, from Goodfellow et al. [2016] explains distributed representation with a simple example:

> For example, suppose we have a vision system that can recognize cars, trucks, and birds, and these objects can each be red, green, or blue. One way of representing these inputs would be to have a separate neuron or hidden unit that activates for each of the nine possible combinations: red truck, red car, red bird, green truck, and so on. This requires nine different neurons, and each neuron must independently learn the concept of color and object identity. One way to improve on this situation is to use a distributed representation, with three neurons describing the color and three neurons describing the object identity. This requires only six neurons total instead of nine, and the neuron describing redness is able to learn about redness from images of cars, trucks and birds, not just from images of one specific category of objects. [Goodfellow et al., 2016]

During the connectionist movement of the 1980s, several main ideas emerged, some of which stayed essential to today's Deep Learning theory. The first successful use of back-propagation to train deep neural networks, which is still a popular approach to training Deep Learning models, was reported in Rumelhart et al.'s (1985) work. Other equally important contributions, such as LeCun et al.'s (1995) Convolutional Networks for vision and speech recogntion, and Hochreiter and Schmidhuber's (1997) Long Short-Term Memory Network (LSTM) used for sequence modeling tasks, also arose during the connectionist wave.

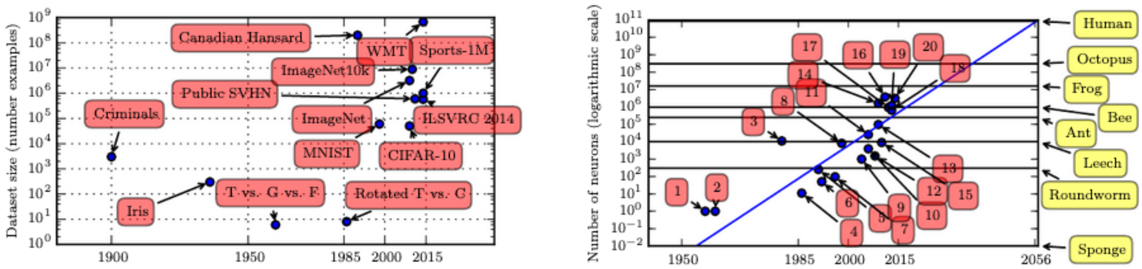
By mid-1990s, neural network-based and other AI-based projects have begun to promise ambitious claims while looking for investments. Investors unhappy when AI's study failed to meet these unrealistic expectations. Neural Networks were too costly to train on real world applications and there were few representative datasets at the time. Simultaneously, other Machine Learning areas started to show progress, achieving good results on many important tasks [Cortes and Vapnik, 1995]. These two factors led to a decline in the popularity of neural networks that lasted until 2007 [Goodfellow et al., 2016].

Deep Learning: 2006–

The most recent, and current, wave of Neural Networks research as re-branded as Deep Learning. In 2006, Geoffrey Hinton introduced the ideas of unsupervised pretraining and Deep Belief Networks (DBN) [Hinton and Salakhutdinov, 2006]. A DBN, contrary to the past approaches, could be efficiently trained using a strategy called greedy layer-wise pre-training. In short, the idea was to train a simple 2-layer unsupervised model, such as a Restricted Boltzmann Machine [Ackley et al., 1985], followed by freezing all of its parameters. Then, a new set of layers is stacked atop, and repeating the same procedure. A deep network is formed by stacking and training layers in a greedy fashion, which can then be used to initialize the parameters of a traditional neural network [Goodfellow et al., 2016].

The term "Deep Learning" was coined as more networks started gaining depth through the greedy layer-wise strategy [Bengio et al., 2007, Marc'Aurelio Ranzato et al., 2007]. By 2011, the speed of GPUs had increased significantly, making it possible to train deep Convolutional Neural Networks such as AlexNet, which achieved astounding results on the ImageNet Large Scale Visual Recognition Competition (ILSVRC) [Krizhevsky et al., 2012].

Deep Learning has since become more popular and useful, largely as a result of more powerful computers, greater amounts of datasets available and advancements in techniques to train deeper networks [Alom et al., 2018]. Figure 14a provides a perspective on the availability data over the years, while Figure 14b reflects how the computing capacity of Neural Networks has grown since its creation. The years to come have many challenges and opportunities to enhance Deep Learning and to take it across new frontiers.



(a) (b)

Figure 14 – Growth of datasets (14a) and of neural network size over time (14b).

Source: Goodfellow et al. [2016]

3.2 Deep Learning in Society

Artificial Intelligence based solutions have the potential to bring huge benefits to society. Deep Learning, in specific, is re-configuring access to information [Wang et al., 2018a], driving [Chen et al., 2015] and even medical diagnosis [Suk et al., 2014]. Contrary to the dystopian, AI-dominated future commonly portrayed in the media [Geraci, 2012], the immediate concerns caused by this fast diffusion of Deep Learning in the daily lives of millions of people revolve around fairness, privacy and transparency. In a SLR system, these factors are of primary importance. For instance, sign languages are not ethnically exclusive, hence, a SLR system should be *fair* in the sense that it is indifferent to skin tone. Moreover, sign translation frameworks should be immune to external interference, conveying, as precisely as possible, the meanings that the signaler intended to transmit. Finally, SLR systems should be *transparent*, establishing a layer of human-machine confidence, and thus, permitting communication to flow as naturally as possible. This section explores each of these concerns, aiming to understand how AI solutions can be designed to with "society-in-the-loop".

3.2.1 Fairness

Machine learning systems are increasingly influencing each facet of human life, including the quality of health care and education that someone receives [Caruana et al., 2015, Bosch et al., 2016], the news or social media, who is given a job, who is released from prison and who undergoes increased policing [Chouldechova and Roth, 2018]. This growth has brought attention to ML's potential to increase social inequities in many research communities, as well as in the popular press Holstein et al.'s (2018).

Learning is not merely a memory transfer method. It includes generalizing from examples rather than just memorizing the particular details that occur in the observations. This is the induction method: general rules from particular examples are drawn — rules that take into account past instances efficiently but also apply to future, unknown instances. It is hoped that future instances can be comparable to previous instances, although not precisely the same [Barocas et al., 2018].

This involves providing representative examples for reliably generalizing models for ML from previous observations. It can be achieved, for instance, by providing a sufficient amount of observations to capture subtle patterns; a sufficiently varied dataset, to show the various kinds of appearances that objects might have; and a sufficiently annotated set of examples to provide reliable ground-truth [Barocas et al., 2018]. A learned model is only as reliable as the data on which it was trained, thus, high quality data is critically important to ML. Consequently, when designing socially sensitive applications - such as a sign language translation system - constructing a representative dataset demands close attention.

Cases of ML-based applications reproducing systemic unfair behavior - for example, hiring systems which are more likely to recommend candidates from specific gender or race; or a criminal recidivism predictor that correlates race with higher probabilities of relapse [Chouldechova, 2017] - are not direct indications that the system's designer meant to reproduce social inequalities. It is important to understand when such disparities are, in fact, discrimination. Analyses on whether the observed disparities are justified or detrimental must be performed [Binns, 2017]. These issues seldom have easy answers, but the comprehensive literature on philosophical and sociological discrimination can assist in the reasoning process.

Many different statistical metrics exist to quantify a ML model, such as precision, recall and f-score. None of them require previous knowledge of sociological theory, and are relatively straightforward to measure. On the other hand, attention to fairness criteria in

AI is fairly recent [Chouldechova and Roth, 2018], and mathematical modeling of such metrics still incites heated debates within the research community [Corbett-Davies and Goel, 2018]. While it is difficult to foresee the effect of implementing a fairness criterion as a concrete restriction in ML models, the growing attention to demographic criteria in Statistics and Machine Learning reflects a change in how intelligent systems are being conceptualized and the perception of the responsibilities of those building them.

3.2.2 Privacy and Security

Most of recent advances in Deep Learning were enabled by the collection of large and representative datasets. Massive data collection, however, presents obvious privacy issues. Ownership of highly sensitive data, such as users' photos and recordings, is taken by companies that collect it. Furthermore, users can neither delete the data generated by them nor restrict the purpose for which such data is used [Shokri and Shmatikov, 2015].

Artificially intelligent systems built on top of these massive datasets have become one of the inseparable technologies in today's world. Highly sensitive services, such as autonomous driving and medical diagnosis are benefiting from advancements in DL research. Thus, understanding the problems of security and privacy that revolve around AI systems can no longer be overlooked. To address this issue, a study by Bae et al. [2018] surveys the current methods proposed to enable robust AI systems. The authors define the notion of SPAI: Secure and Private AI.

Secure AI focuses on attacks and defense on AI systems. In terms of Deep Learning, a system constitutes a model that is learned from the data available. Bae et al. [2018] addresses two major types of security attacks: evasion and poisoning attacks. A poisoning attack takes part in the training stage and attempts to subvert the model during learning. On the other hand, if adversarial observations are used in the inference stage to deliberately lead the model to misclassify the input, this attack is called an evasion attack.

The second notion explored, Private AI, aims for AI systems that preserve data privacy. Because of computing costs or the need for collaborative training, Deep Learning systems may require transferring users' sensitive information to distant computers. In such situations, after the transfer, users lose control over the data and have concerns about their data privacy being stolen between transfers, or that their data may be misused without consent.

Although defense methods [Buckman et al., 2018, Gu et al., 2018, Sun et al., 2018] and privacy-preserving techniques [Shokri and Shmatikov, 2015, Abadi et al., 2016] have been recently proposed, works in the area are still in relatively early stages. To guarantee robust, deployable SPAI systems, it is interesting that more studies are performed around the different types of attacks AI systems are subject to. Research on defense and privacy mechanisms are equally important, taking into account practical considerations on processing time and throughput.

3.2.3 Interpretability

As Deep Learning models become widespread key fields, for instance, in medicine, criminal justice, and financial markets, it seems problematic for people to be unable to understand these models [Caruana et al., 2015]. Although black-box DL systems in place nowadays provide the end user with strong predictive power, they are generally abstruse, in a way that creates room for distrust [Lipton, 2016].

Enabling human-machine confidence should become a necessary objective, especially in sensitive applications, such as automated medical diagnosis [Bhatt et al., 2019]. In other words, system design should take into account training interpretable models or coupling black-box with explainable models, demystify the reasoning process while preserving respectable precision rates [Bhatt et al., 2019]. To this end, Lipton [2016] defines the two key concepts that enable the identification of interpretability problems and system properties that either enhance or compromise human interpretation of Machine Learning models.

The first is interpretability through *transparency*. Transparency connotes some level of understanding the mechanism by which the model operates. Three levels of transparency are considered: at the entire model, or simulatability; at individual components, e.g. parameters, or decomposability; and at the level of the training algorithm, or algorithmic transparency.

Model interpretability can also be achieved post-hoc, that is, extracting information from learned models. Common ways of enabling post-hoc interpretations is through coupling a black-box model with visualizations of learned representations and natural language exaplanations. While this approach may not precisely elucidate the way how a model works, it nonetheless may confer important information to the end user. Post-hoc interpretability is akin to the level that humans can be considered interpretable, since the processes by which decision making happens and the reasoning behind such processes may be distinct.

Chapter 4

Deep Learning Applications

This chapter presents two applications of Deep Learning which highlight some of the advancements achieved in the realm of Computer Vision. The first concerns the ability to semantically understand an unknown image, recognizing the different objects present in the scenario. The second application, human pose estimation, is largely applicable to sign language recognition systems, and is generally used to segment the different body parts such as the hands and face. The two applications are complementary, enabling both the identification of humans in images and the localization of specific regions of interest for sign and action recognition tasks.

4.1 Semantic Segmentation

Semantic segmentation sparks interest in several areas, including human-computer interaction [Zuo, 2016], medicine [Desai et al., 2019] and gesture recognition [Dadashzadeh et al., 2018]. Accordingly, literature on the topic is extensive. Semantic segmentation is one of the high-level tasks that allows the development of applications with high inference capacity of knowledge about the content of an image [Guo et al., 2018]. Figure 15 illustrates different computational tasks that can be performed to automatically parse visual information.

The aim of segmentation algorithms is to assign every pixel in the image to semantically similar groups. Therefore, semantic segmentation requires correct and precise detection of all objects in an image, which, in turn, is a very challenging task in Computer Vision. An example of semantically partitioning an image into cohesive sub-regions can be seen in Figure 16. Human segmentation, as seen in Figure 16, is particularly interesting for SLR systems as it can help reduce the visual complexity of a scene by cropping the region where the signaler appears.

Studies in the field date back to 1970's, when methods were mostly comprised of traditional Computer Vision techniques, such as edge detection and *thresholding* [Fu and

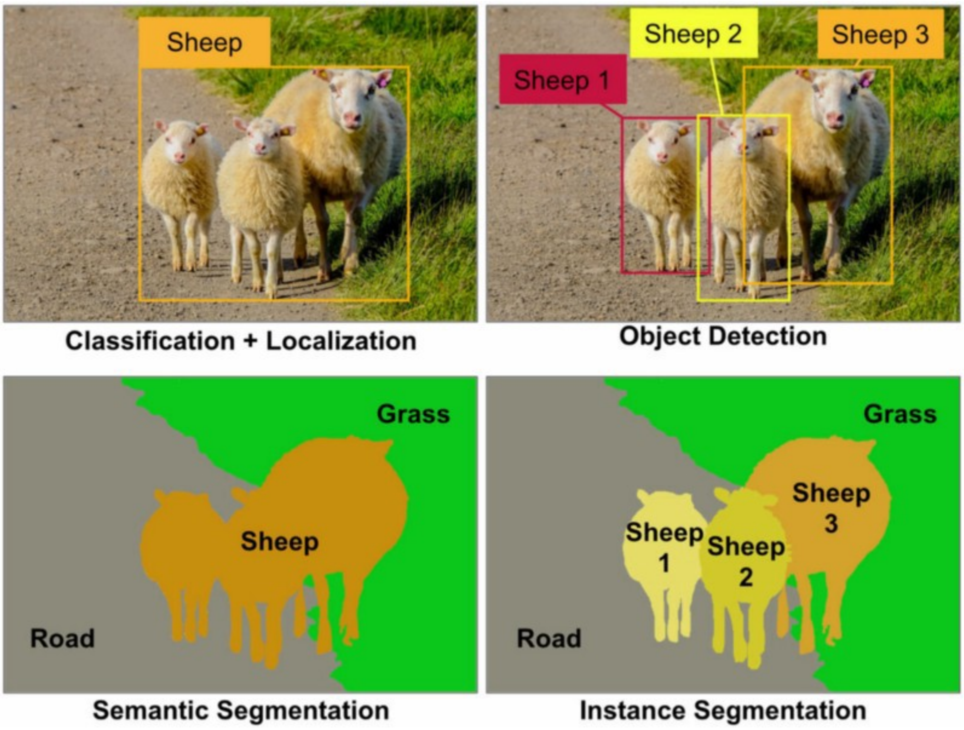


Figure 15 – Comparison among different segmentation tasks. Classification identifies the presence or absence of an object in a given image. Semantic segmentation identifies which pixels belong to a class of objects. Object detection identifies in which region of the image an object appears. Last, instance segmentation provides pixel-wise masks of class of objects along with the identification of each individual instance.

Source: Parmar [2018]

Mui, 1981]. Recently, with the climbing popularization of Deep Learning techniques, many of the problems of semantic segmentation have been addressed through DL architectures. In particular, Convolutional Neural Networks have been obtaining superior results in terms of accuracy and efficiency to other methods traditionally applied [Girshick et al., 2013, Long et al., 2015, Mazzini et al., 2018].

Regions with CNN, or simply R-CNN, proposed by Girshick et al. [2013] outperformed by 30% the previous best results on PASCAL Visual Object Classes (VOC2012) challenge [Everingham et al., 2012], a benchmark for object detection containing over 27 thousand annotated objects in 11,530 images.

R-CNN is comprised of three main modules, as illustrated in Figure 17. The first generates 2000 class-indifferent region proposals for all objects in the image through *selective search* [Uijlings et al., 2013]. The second performs feature extraction using Krizhevsky et al.'s (2012) pre-trained CNN, obtaining a 4096-dimensional feature vector from each region proposal. Finally, a set of class-specific linear SVMs identify the object enclosed within each proposed region.

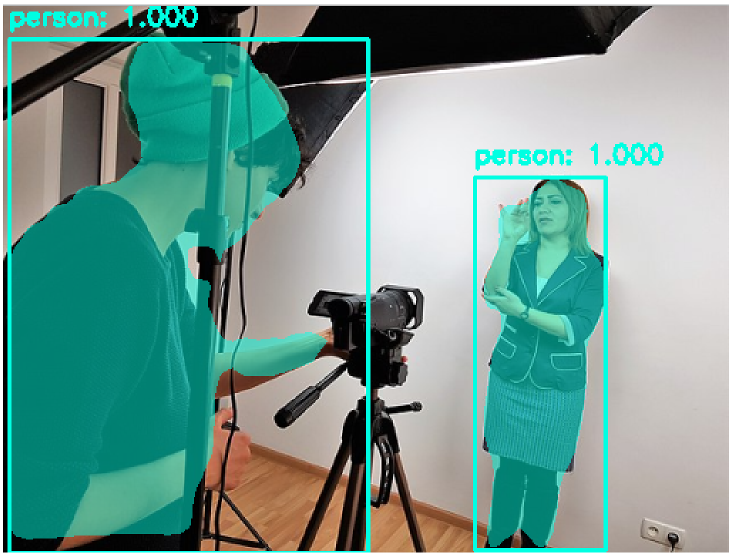


Figure 16 – Semantic segmentation applied to identify people in images. In this particular example, SS can be used to identify humans in sign language videos, ignoring visual information that is irrelevant to sign recognition tasks.

Source: Dato [2018]

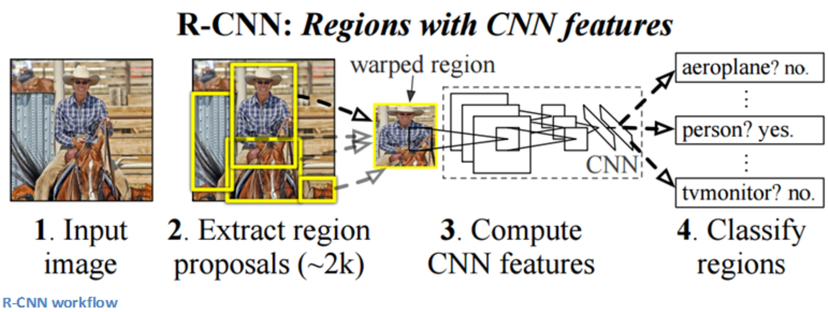


Figure 17 – R-CNN system overview

Source: Girshick et al. [2013]

A major drawback of R-CNN is the large amount of time spent to train the network, since there are over 2000 regions per image. Also, selective search is a fixed method where no learning stage is involved, potentially leading to the generation of bad candidate regions. In this context, Girshick [2015] proposed *Fast-CNN*, integrating the region proposal stage directly into the CNN pipeline. Fast R-CNN is faster than R-CNN since the convolutional network itself generates a feature map directly from each image, eliminating the costly convolutional step over 2000 image segments. Selective search is performed over the feature maps to generate region proposals. Moreover, Fast R-CNN also absorbed the classification stage by replacing the set of linear SVMs by a softmax layer, which predicts each class from resized proposed regions.

Because Fast R-CNN uses selective search to identify region proposals, this stage still affects the general performance of the network. Therefore, Ren et al. [2015] developed the Faster R-CNN, which lets the network itself learn the region proposals. This new version enables almost real time testing, as seen in Figure 18.

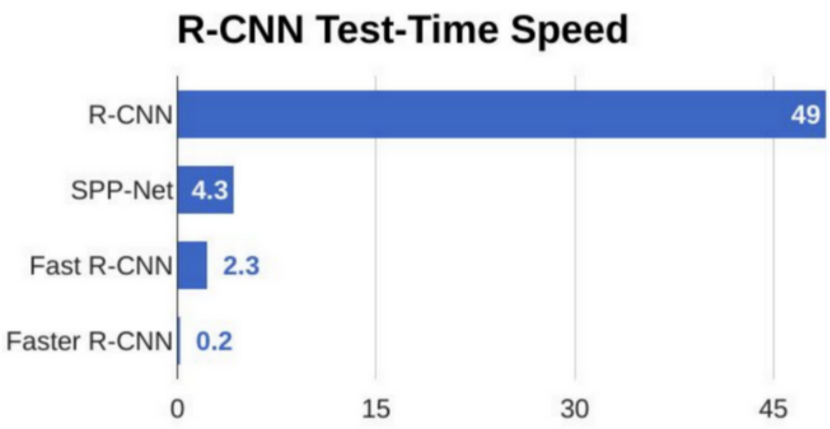


Figure 18 – Comparison among testing times (in hours) of R-CNN, Fast R-CNN and Faster R-CNN

Source: Gandhi [2018]

The importance of this advancement connects back to SLR, insofar as the speed of sign recognition can greatly impact the flow of communication in a natural setting. Therefore, an ideal layer of segmentation to reduce visual complexity should not affect the performance of the system as a whole.

Last, He et al. [2017] proposed an approach called Mask R-CNN, which extends Faster R-CNN by including a parallel branch for predicting objects' masks. A diagram of Mask R-CNN and Faster R-CNN architectures can be seen in Figure 19.

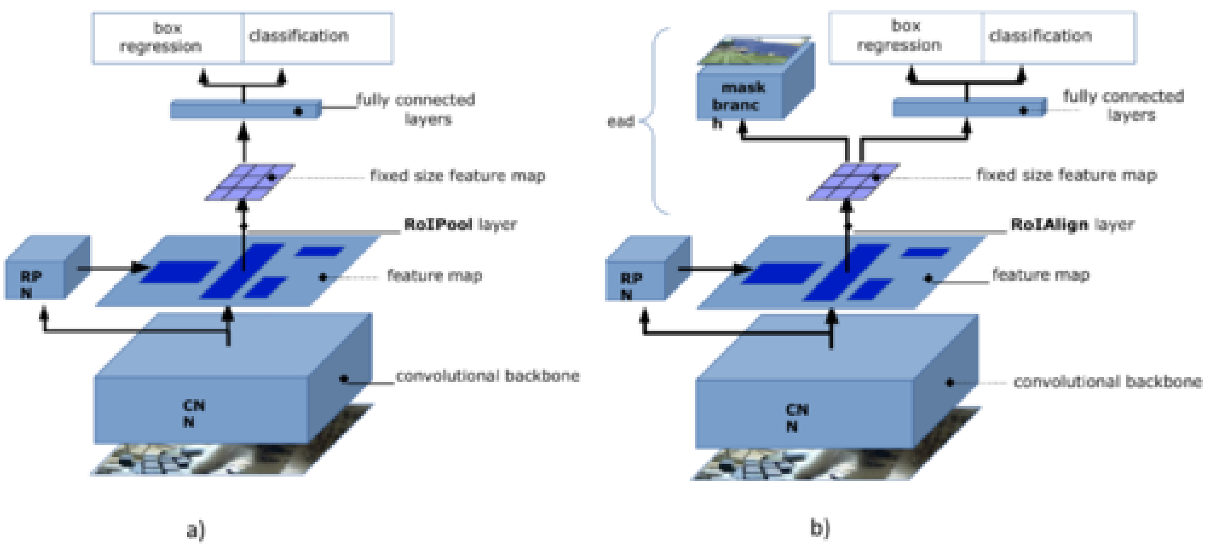
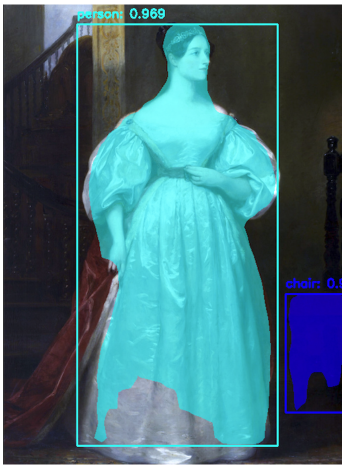


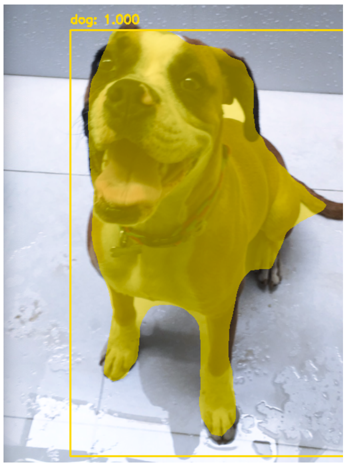
Figure 19 – Faster R-CNN (a) and Mask R-CNN (b) architectures.

Source: Lim [2017]

An example of semantic segmentation using Mask R-CNN can be seen in Figure 20. As discussed, Mask R-CNN is able to both perform object detection and generate pixel-wise masks of each recognized instance. Figure 20 shows a few examples of Mask R-CNN's results obtained using a model pre-trained on the COCO dataset [Lin et al., 2014, Rosebrock, 2019].

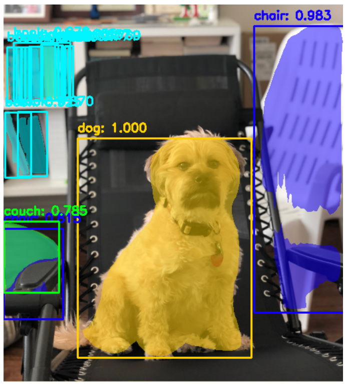


(a) Ada Lovelace[1]



(b) Oliver

(c) Baymax



(d) Yoshi

Figure 20 – Examples of semantic segmentation using Mask R-CNN

One important observation regarding the application of Mask R-CNN to aid in sign language recognition systems is that, because masks are generally not precise, important areas of the image could be left out. This particular effect can be seen in Figure 20a: the mask covers most of the significant areas for sign recognition, nonetheless, part of the right hand was not captured. In the context of SLR, the absence of information concerning the fingers could reflect in the misclassification of a sign. Therefore, a possible solution to this problem could be to segment the image with respect to the smallest bounding box that encloses an object, in place of its bit-wise mask.

The next section presents another problem that can greatly benefit from the advancements in semantic segmentation. In a like manner of the example shown in Figure

[1]Source: Carpenter [1836]

16, semantic segmentation can serve as a pre-processing step to pose estimation systems, identifying humans on images and excluding other visual information.

4.2 Human Pose Estimation

The major goals of human pose estimation (HPE) are mapping human joints, such as elbows and writs, and identifying the different parts of the body in images and videos. From medicine [Obdržálek et al., 2012] to video-games, human pose estimation enables a variety of applications. Despite having been an active research topic for many years now [Hogg, 1983, Xiao et al., 2018], HPE still challenges the Computer Vision community. Pfister [2015] lists the main difficulties:

(i) high variability in human body shapes

(ii) high variability of human appearance due to lighting, viewing angle, clothing and background

(iii) high dimensionality of possible poses

(iv) ambiguities due to the loss of depth information in 2D images

(v) motion blur

(vi) occlusions

Some examples of the challenges listed above can be seen in Figure 21.

Figure 21 – Examples of challenging setups for human pose estimation

Source: Pfister [2015]

Human pose estimation was initially addressed by directly modeling characteristics of the human body. For instance, Hogg [1983] represented a person by hierarchical levels (a person has an arm has a lower-arm, which in turn has a hand), while Forsyth and Fleck [1997] established a "body plan" for people and for animals. The basic idea behind the classical approaches is to describe the body as a deformable configuration, as Figure 22 shows. A person is represented by a collection of parts that can parameterized by pixel location and orientation. This collection is then matched against pre-defined templates, identifying possible valid configurations.

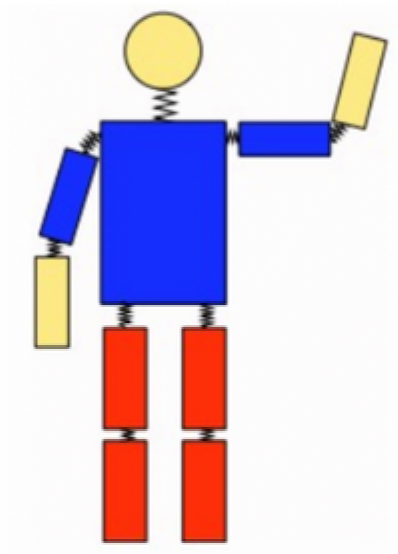


Figure 22 – Classical representation of humans in pose estimation problems

Source: Babu [2019]

Although classical approaches have evolved to allow for complex configurations, as in Yang and Ramanan [2012], model-based approaches are limited in expressiveness. Evidently, only a finite number of different pose templates can be defined. After "DeepPose", proposed by Toshev and Szegedy [2014], HPE research began to steer towards Deep Learning. Convolutional networks have replaced the labor-intensive stage of defining templates, whilst yielding outstanding improvements on standard benchmarks. A study by Wei et al. [2016], for instance, was able to estimate non-standard and ambiguous poses with the use of *Convolutional Pose Machines (CPMs)*, a series of CNNs that generate 2D mappings at each image location. The results of CPMs on three different benchmark datasets can be seen in Figure 23.

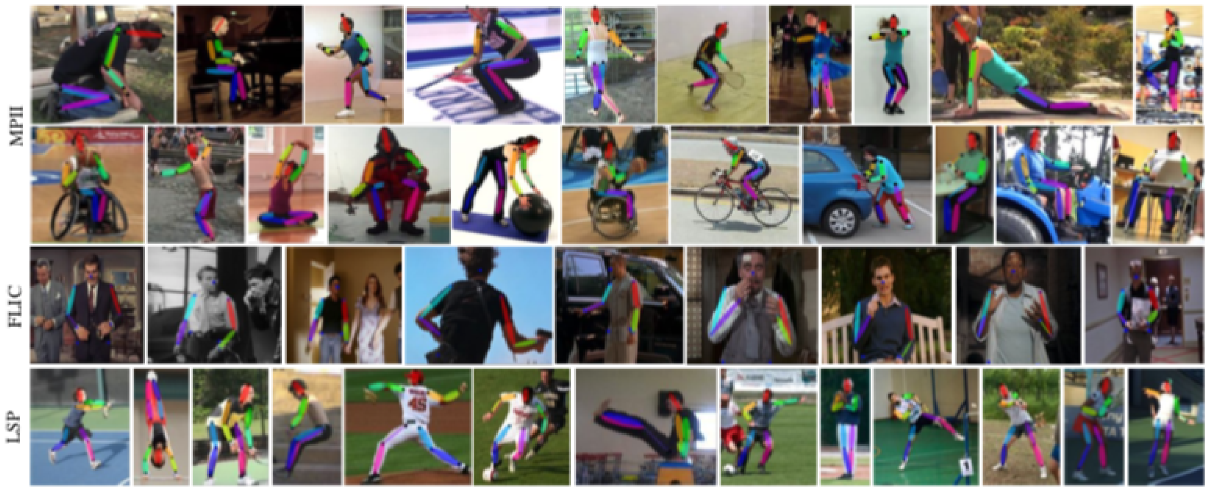


Figure 23 – Non-standard and ambiguous poses estimated using CNNs

Source: Wei et al. [2016]

Likewise, Gattupalli et al. [2016] introduced an interesting example of HPE using Deep Learning. The study explores CNNs for pose estimation on an annotated sign language dataset, so as to provide useful features for sign language recognition tasks. As discussed previously, sign languages are composed of a multitude of manual - hand configuration and orientation - and non-manual - such as facial expression and body posture - parameters. In contrast with manual-focused approaches, few studies have been performed to utilize the non-manual features of the language [Er-Rady et al., 2017]. Meanwhile, body posture, which can be obtained through HPE techniques, can be beneficial to aid recognition in signs that involve movement on a certain body location, as seen in Figure 24. Moreover, HPE can also help distinguish between sign language dialogues and stories by observing changes in body positions when the signaler addresses different interlocutors [Gattupalli et al., 2016].

(a) Bruise (b) A bruise around one's eye

Figure 24 – Example of a ASL signs that involve movement in specific body regions. 24a shows the final configuration for the sign "bruise", while 24b denotes "bruise" around the eye region.

Source: Lapiak [2019]

Chapter 5

Extracting Regions of Interest

This chapter presents training results of a Mask R-CNN architecture applied to ROI extraction. As discussed in Chapter 4, Mask R-CNN is an efficient alternative to previous object detection methods, with the added capability of performing pixel-wise labeling. Section 5.1 elaborates on the implementation of Mask R-CNN used in this project. Section 5.2 discusses the main sources of training data for this project. Section 5.3 discusses the experiments and results obtained using the architecture.

5.1 Mask R-CNN Configuration

The implementation of Mask R-CNN used in this thesis is the one provided in Abdulla [2017] and can be divided in four main stages, as follows.

5.1.1 Backbone

The first corresponsds to the backbone, a standard convolutional neural network taht serves as a feature extractor. In this implementation, the backbone is a combination of ResNet101 and the Feature Pyramid Network (FPN) [Lin et al., 2016]. The FPN adds a second sequence of feature extraction layers that passes high level features from the first convolutional pyramid down to the lower layers, as show in Figure 25. This allows the diffusion of lower and higher level features in every level of the network.

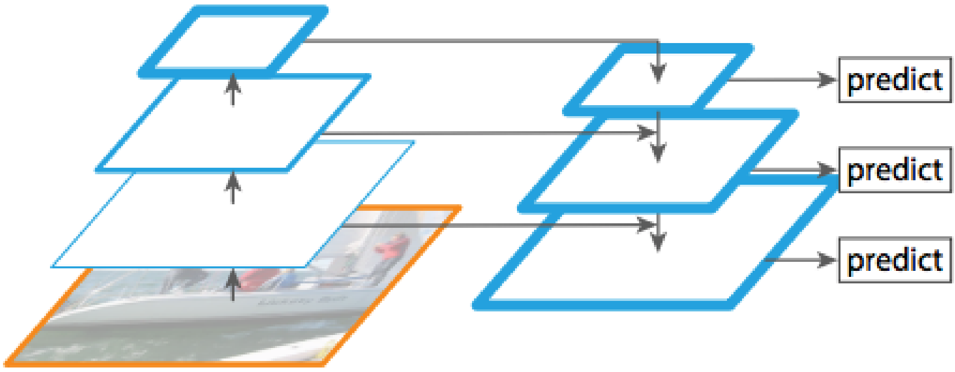


Figure 25 – Feature Pyramid Networks

Source: Lin et al. [2016]

5.1.2 Region Proposal Network (RPN)

As discussed in Chapter 4, Mask R-CNN contains a second level of CNNs that scan the image to locate possible objects. The scanned regions are called *anchors* and correspond to distributed boxes over the image area, as shown in Figure 26. An RPN scans about 200 thousand overlapping anchor boxes of different sizes and aspect ratios to maximize the coverage area.

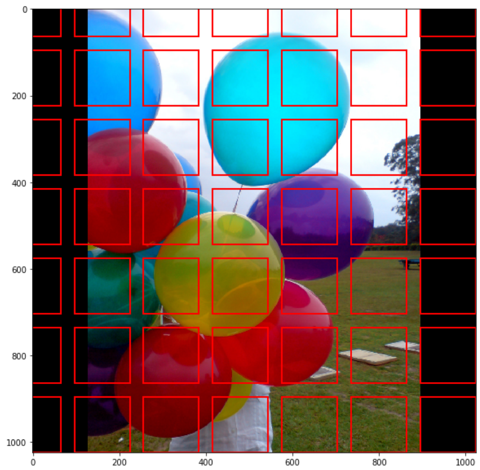


Figure 26 – Illustration of anchor boxes scanned in the region proposal network layer. During the pipeline, RPN scans over the feature map generated by the backbone, instead of directly over the input image.

Source: Abdulla [2018]

The RPN classifies each anchor as foreground - likely to enclose an object - or background. Foreground anchors are refined to better capture the center of an object. In case multiple anchors overlap, only the ones with the highest foreground score are kept, a mechanism referred to as Non-max Suppression. The final region proposals are then passed to the next stage.

5.1.3 ROI Classifier and Bounding Box Regressor

The ROI classifier acts similarly to the RPN. For each region proposal, the classifier outputs the specific class label of the object in the ROI, or marks the region as *background*, for discarding. Each foreground bounding box is further refined to encapsulate the object. Both the class label and the bounding box are passed to the final stage of the architecture.

Although the classifiers used in this stage generally require a fixed input size, region proposals outputted in the RPN stage can present varying sizes. To deal with this inconsistency, the original Mask R-CNN architecture contained a method called ROIAlign, in which samples of different points of the feature map are bi-linearly interpolated to generated fixed sizes. In Abdulla's (2017) implementation, ROIAlign was replaced by a function to crop the desired part of the image and then resize it.

5.1.4 Segmentation Masks

Last, the mask branch is a convolutional network that takes the positive regions identified by the ROI classifier and produces masks for them. This layer represents the novelty introduced by Mask R-CNN in relation to Faster R-CNN. The masks produced are of low resolution (28x28 pixels) to optimize calculations in this branch, as show in Figure 27. They are also made of float numbers, holding more information than standard binary masks. Ground-truth masks are scaled down during training to compute loss, while during inference masks are scaled up to the size of the ROI bounding box, outputting a final mask per object.

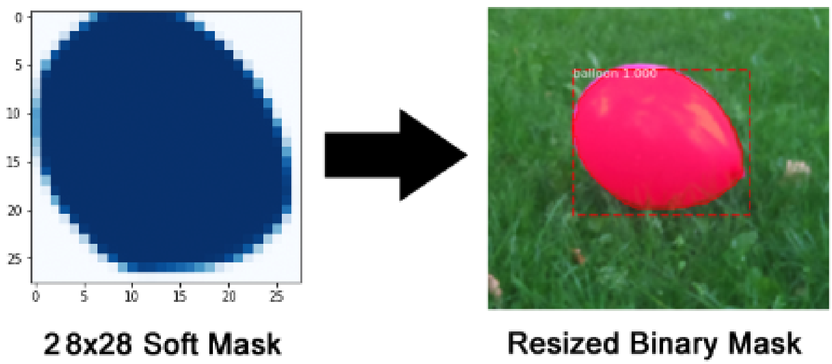


Figure 27 – Illustration of a soft mask generated by Mask R-CNN

5.2 Data sets

In Deep Learning, the quality and quantity of training data are determining factors to a model's performance. Thus, searching for representative data sets is a crucial step towards achieving a satisfactory extractor. Despite the abundance of human key-point annotations for pose estimation, such as the ones shown in Figure 23, few data sets provide masks that can be used to train segmentation approaches. While technologies such as Microsoft's Kinect™ can annotate joint key-points in near real-time, semantic labeling is labor intensive.

This section describes two resources available online that can be useful to developing body parsing applications. Due to the nature of the application proposed in this project, the data sets selected contain both "face" or "head" and "hand" or "lower arm" labels.

5.2.1 PASCAL-Part Dataset

The Pattern Analysis, Statistical Modelling and Computational Learning Visual Object Classes (PASCAL-VOC) provides standardized image data sets for object recognition Everingham et al. [2012]. PASCAL-VOC 2012, the latest release, contains 11,530 images for training and validation, accompanied by 27,450 ROI annotated objects and 6,929 segmentations of 20 semantic classes.

Chen et al. [2014] extended the 2010 edition of PASCAL-VOC to include segmentation masks for each body part of the object, as seen in Figure 28. Despite being one of the largest available data sets for human body parsing, PASCAL-Person-Part contains less than 2,000 labeled images for training. This amount is significantly smaller than the size of common benchmarks for image classification, such as COCO[1] and ImageNet[2].

Figure 28 – Examples of PASCAL-Part mask annotations

Source: Chen et al. [2014]

5.2.2 Weakly and Semi-Supervised Human Body Part Parsing (WSHP)

Fang et al. [2018] proposes a methodology for transferring human body part key-point annotations to raw images by exploiting anatomical similarities among different persons with similar poses. As a result, more training data can be generated in a weakly

[1]200 thousand labeled images
[2]14 million annotated images

supervised manner, allowing the expansion of existing key-point data sets to semantic segmentation problems.

Although the methodology generates less precise masks, as shown in Figure 29, the authors were able extended PASCAL-Person-Part to 3.5 thousand labelled instances using WSHP. Moreover, the authors have also applied semi-supervised labelling to the Multi Human Parsing II [Zhao et al., 2018] and COCO Keypoint data sets, amounting to a total of 150 thousand images. The final data set contains annotations of head, torso, upper arms, lower arms, upper legs and lower legs, with no distinction of left and right.

Figure 29 – Examples of annotations obtained with the weakly and semi-supervised human body parsing method

Source: Fang et al. [2018]

5.3 Results and Discussions

All experiments were conducted in Google Cloud Compute Engine VM, using 8 vCPUs with 30GB of memory and a single NVIDIA Tesla K80 GPU. Data augmentation was performed online in both experiments, and encompassed either horizontal mirroring, 90^o or 180^o rotation, or Gaussian blurring. The training schedule consisted of:

1. 30 epochs of training network heads, starting with weigths pre-trained on the COCO data set;

2. 10 epochs of fine tuning layers 4 and up.

The total training time was of 13 hours for PASCAL-Person-Part and over 52 hours for WSHP. For each experiment, the mean Average Precision (mAP) at a value of Intersection over Union (IoU) of 0.5 was calculated. The Average Precision calculates the average percentage of correct predictions of a class over values of recall ranging from 0 to 1. Because location is important in a segmentation task, the Intersection over Union measures the overlap between a predicted boundary with the ground truth label. Defining a value of IoU to 0.5 represents that the predicted mask is correct if it intersects the ground truth label in at least 50% of its total area. Hence, the mean Average Precision is mean value of AP calculated for all classes in the dataset, considering the prediction threshold.

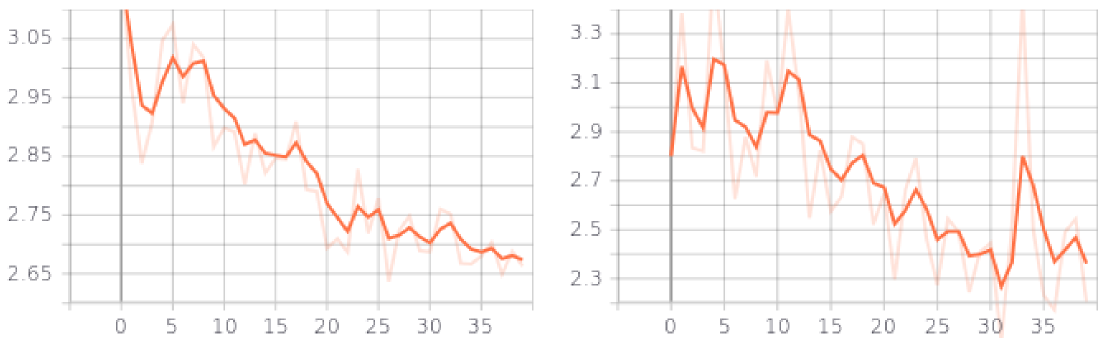
The models were also validated in a set of 100 randomly selected frames from videos of Brazilian Sign Language signs, as illustrated by Figure 30, available at the Libras-20 data set [Almeida et al., 2019]. Since the images are not annotated, the results were compared by manual inspection. In this case, the model accuracy was determined by the percentage of the frames in which the model was able to correctly identify all regions of interest.

(a) (b) (c)

Figure 30 – Sample frames extracted from Libras-20 data set

Source: Almeida et al. [2019]

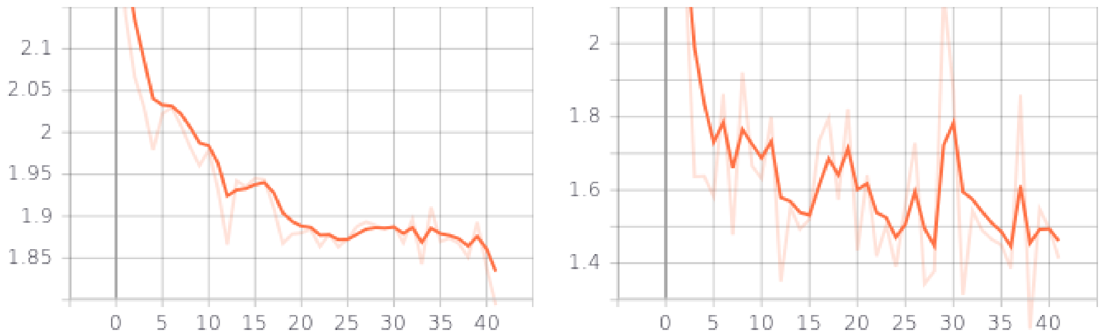
Before starting the training pipeline, the PASCAL-Person-Part data was pre-processed to adequate mask labels to the WSHP standard. The training and validation loss plots for PASCAL-Person-Part can be seen in Figure 31. A $\mathrm{mAP}_{IOU} = 0.5$ score of 19.87 ± 4.56 was obtained after performing 20 rounds of inference over test sets of 40 images.



(a) Loss (b) Validation loss

Figure 31 – Training and validation loss while for PASCAL-Part data

The training and validation loss plots for WSHP can be seen in Figure 32. Due to resource limitations, this experiment used only 20 thousand of the original 150 thousand images for training. As expected, a greater quantity of training data helped the model to achieve lower losses in both training and validation stages. The enhancement was also reflected in mAP$_{IOU=0.5}$ score, which increased to 32.09 ± 2.14 points, under the same inference conditions as the last experiment.



(a) Loss (b) Validation loss

Figure 32 – Training and validation loss for WSHP data

A comparison of both models performance on Libras-20 data set can be seen in Figure 33.

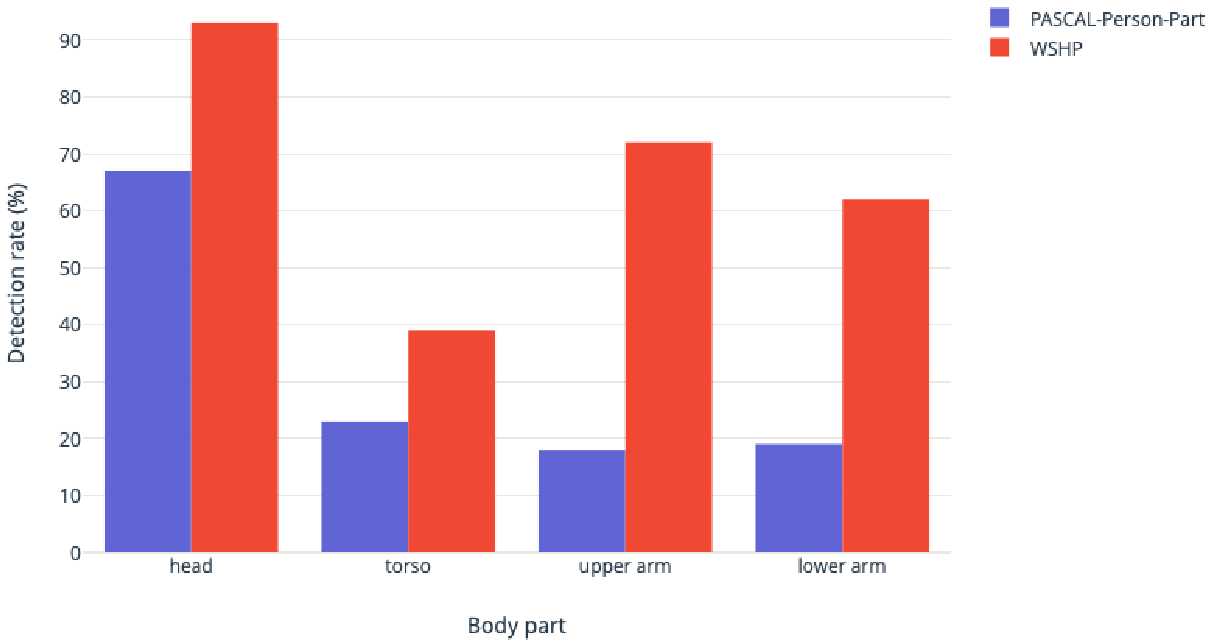
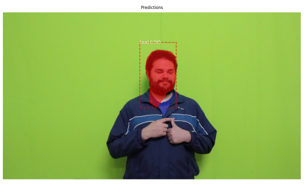
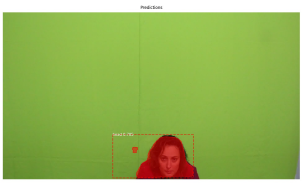


Figure 33 – Detection rate of Mask R-CNN models applied to Libras-20 frames

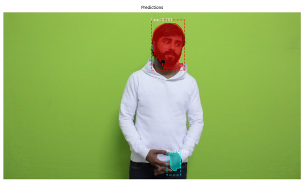
Analyzing the results, both models presented highest detection rates for the facial region. In general, PASCAL-Person-Part was unable to detect the lower arms. Sample outputs of the model trained with PASCAL-Person-Part can be seen in Figure 34.



(a) Incomplete



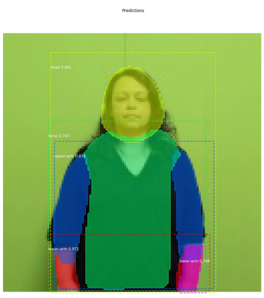
(b) Incorrect



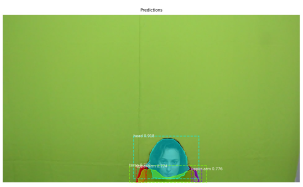
(c) False positive

Figure 34 – PASCAL-Person-Part Mask R-CNN detection on Libras-20. 34a Incomplete: Model was unable to identify non-facial regions. 34b Incorrect: Mask boundaries encompassed incorrect regions of the body. 34c False positive: Model identified lower arm region as head.

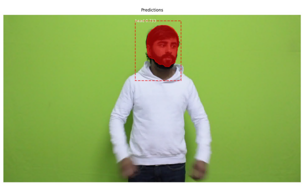
As seen in Figure 33, adding weakly labeled data to the training set enabled arm and torso detection, as observed in Figure 35, however, detection masks still lose information around fingers or fails to detect arms in clear situations, such as 35c.



(a) Correctly segmented



(b) Correctly segmented



(c) Incomplete

Figure 35 – WSHP Mask R-CNN detection on Libras-20. 35a presents a complete segmentation output. 35b presents a refinement of segmentation masks in relation to PASCAL-Person-Part model. 35c presents an incomplete output, despite segmentation targets being clearly visible in the image.

Chapter 6

Final Remarks

Deep Learning theory has evolved significantly since it was first proposed in the 1940s, under the title of *Cybernetics*. Cutting-edge learning techniques, coupled with unprecedented processing power, and massive amounts of data enable the design of solutions to problems once thought too complex solve. Artificial Intelligence is revolutionizing the way people live, interact and work.

In this context, the first half of this thesis aimed to understand how Deep Learning can be used to bridge the communication gap between the deaf and the hearing communities. The first step towards designing a robust sign language recognition system using Deep Learning is understanding the theory behind DL itself. This was achieved in three stages. First, the fundamentals of DL theory were laid out. Then, two specific applications of DL architectures that are important to sign language recognition were presented. Finally, DL was discussed under historical and social perspectives. Study in all three areas represented key components to the progression of this work.

In the second half of this project, Mask R-CNN was implemented and trained to detect regions of interest in sign language videos, aiming to reduce visual complexity and improve sign recognition. Two experiments were performed. The first comprised of training the architecture on PASCAL-Person-Part 2000 instances. Likely due to the scarcity of images, the model presented low performance, achieving a $\text{mAP}_{IOU=0.5}$ of 19.67 ± 4.56, below the baseline mAP[1] for part appearance presented in Gonzalez-Garcia et al. [2018].

In a second experiment, Mask R-CNN was trained using weakly labeled data generated by transferring key-point annotations to raw images. Due to resource limitations, this experiment used only 20 thousand of the original 150 thousand images for training. Nonetheless, this approach allowed a $\text{mAP}_{IOU=0.5}$ score increase to 32.09 ± 2.13 points. Also, when tested on the Libras-20 frames, the model was able to detect the regions of interest with a mean detection rate of 66.3%.

[1] mAP=22.1

6.1 Future Work

There are several areas for future work:

- **Train Mask R-CNN with the full WSHP data set:** Mask R-CNN can be retrained with all 150 thousand labelled instances of the WSHP data set. Performance improvements could also be achieved by testing different hyper-parameters.

- **Train networks separately for each region:** EgoHands [Bambach et al., 2015], Rendered Handpose Dataset [Zimmermann and Brox, 2017] and Wider Face [Yang et al., 2016] provide extensive data for hands, fingers and face detection in the wild, respectively. Mask R-CNN or different architectures could also be specialized to identify each region of interest in separate flows, improving local precision.

- **Train with different architectures:** Research in Deep Learning is evolving at a fast pace, and new architectures have been released since the development of this project. Alternatives such as Gated Shape CNNs for Semantic Segmentation [Takikawa et al., 2019] - which wires shape information in a separate flow, reinforcing masks' boundaries - and FastFCN [Wu et al., 2019], which runs three times faster than ResNet as a backbone, could be further studied.

- **Implement the ROI extraction module to the aid in sign language recognition:** This would involve adding the trained Mask R-CNN model to the sign language recognition system, as the one proposed in Chapter 1.

References

M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318. ACM, 2016.

W. Abdulla. Mask r-cnn for object detection and instance segmentation on keras and tensorflow. https://github.com/matterport/Mask_RCNN, 2017.

W. Abdulla. Splash of color: Instance segmentation with mask r-cnn and tensorflow, Dec 2018. URL https://engineering.matterport.com/splash-of-color-instance-segmentation-with-mask-r-cnn-and-tensorflow.

D. H. Ackley, G. E. Hinton, and T. J. Sejnowski. A learning algorithm for boltzmann machines. *Cognitive science*, 9(1):147–169, 1985.

A. Ahmed, Y. Hifny, K. Shaalan, and S. Toral. End-to-end lexicon free arabic speech recognition using recurrent neural networks. *Computational Linguistics, Speech And Image Processing For Arabic Language*, 4:231, 2018.

G. T. B. Almeida. Criação de banco de sinais de libras para implementação de sistemas com visão computacional. Bachelor's thesis, UFMG, 2017.

S. Almeida. *Extração de Características em Reconhecimento de Parâmetros Fonológicos da Língua Brasileira de Sinais utilizando Sensores RGB-D*. PhD thesis, UFMG, 2014.

S. G. M. Almeida, F. G. Guimarães, and J. A. Ramírez. Feature extraction in brazilian sign language recognition based on phonological structure and using rgb-d sensors. *Expert Systems with Applications*, 41(16):7259–7271, 2014.

S. G. M. Almeida, T. M. Rezende, G. T. B. Almeida, A. C. R. Toffolo, and F. G. Guimarães. Libras-20 dataset, July 2019. URL https://doi.org/10.5281/zenodo.2667329.

M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, B. C. Van Esesn, A. A. S. Awwal, and V. K. Asari. The history began from alexnet: a comprehensive survey on deep learning approaches. *arXiv preprint arXiv:1803.01164*, 2018.

A. Amidi and S. Amidi. Cs 230 - convolutional neural networks cheatsheet. Lecture Notes, Nov 2018. URL https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks.

S. C. Babu. Online, Apr 2019. URL https://blog.nanonets.com/human-pose-estimation-2d-guide/. [Accessed May 31st, 2019].

H. Bae, J. Jang, D. Jung, H. Jang, H. Ha, and S. Yoon. Security and privacy issues in deep learning. *arXiv preprint arXiv:1807.11655*, 2018.

S. Bambach, S. Lee, D. J. Crandall, and C. Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

S. Barocas, M. Hardt, and A. Narayanan. *Fairness and Machine Learning*. fairmlbook.org, 2018. http://www.fairmlbook.org.

A. L. Beam. Deep learning 101 - part 1: History and background. Online, 2017. URL https://beamandrew.github.io/deeplearning/2017/02/23/deep_learning_101_part1.html. [Accessed June 10th, 2019].

Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle. Greedy layer-wise training of deep networks. In *Advances in neural information processing systems*, pages 153–160, 2007.

U. Bhatt, P. Ravikumar, and J. Moura. Building human-machine trust via interpretability. 2019.

R. Binns. Fairness in machine learning: Lessons from political philosophy. *CoRR*, abs/1712.03586, 2017. URL http://arxiv.org/abs/1712.03586.

N. Bosch, S. K. D'Mello, R. S. Baker, J. Ocumpaugh, V. Shute, M. Ventura, L. Wang, and W. Zhao. Detecting student emotions in computer-enabled classrooms. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, pages 4125–4129. AAAI Press, 2016. ISBN 978-1-57735-770-4. URL http://dl.acm.org/citation.cfm?id=3061053.3061231.

J. Buckman, A. Roy, C. Raffel, and I. Goodfellow. Thermometer encoding: One hot way to resist adversarial examples. *International Conference on Learning Representations*, 2018.

A. Canziani, A. Paszke, and E. Culurciello. An analysis of deep neural network models for practical applications. *arXiv:1605.07678 [cs]*, May 2016. URL http://arxiv.org/abs/1605.07678. arXiv: 1605.07678.

M. S. Carpenter. *Portrait of Ada Lovelace (1815-1852)*. Goverment Art Collection (UK), 1836. URL https://commons.wikimedia.org/wiki/File:Ada_Lovelace.jpg.

R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pages 1721–1730, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3664-2. doi: 10.1145/2783258.2788613. URL http://doi.acm.org/10.1145/2783258.2788613.

Z. Che, S. Purushotham, K. Cho, D. Sontag, and Y. Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1):6085, 2018.

C. Chen, A. Seff, A. Kornhauser, and J. Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2722–2730, 2015.

X. Chen, R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, and A. Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1971–1978, 2014.

M. J. Cheok, Z. Omar, and M. H. Jaward. A review of hand gesture and sign language recognition techniques. *International Journal of Machine Learning and Cybernetics*, 10 (1):131–153, Jan 2019. ISSN 1868-808X. doi: 10.1007/s13042-017-0705-5.

A. Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, 5(2):153–163, June 2017. doi: 10.1089/big.2016.0047. URL https://doi.org/10.1089/big.2016.0047.

A. Chouldechova and A. Roth. The frontiers of fairness in machine learning. *CoRR*, abs/1810.08810, 2018. URL http://arxiv.org/abs/1810.08810.

S. Corbett-Davies and S. Goel. The measure and mismeasure of fairness: A critical review of fair machine learning. *CoRR*, abs/1808.00023, 2018. URL http://arxiv.org/abs/1808.00023.

C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

A. Dadashzadeh, A. T. Targhi, M. Tahmasbi, and M. Mirmehdi. Hgr-net: A fusion network for hand gesture segmentation and recognition. *arXiv preprint arXiv:1806.05653*, 2018.

Dato. *English: Recording Armenian Sign Language words at WMAM office*. Wikimedia Commons, Nov 2018. URL https://commons.wikimedia.org/wiki/File:Recording_Armenian_Sign_Language_words_at_WMAM_office,_December_2018.jpg.

A. Dertat. Applied deep learning - part 4: Convolutional neural networks. Online, Nov 2017. URL https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2. [Accessed June 10th, 2019].

A. D. Desai, G. E. Gold, B. A. Hargreaves, and A. S. Chaudhari. Technical considerations for semantic segmentation in mri using convolutional neural networks. *arXiv preprint arXiv:1902.01977*, 2019.

A. Er-Rady, R. Faizi, R. O. H. Thami, and H. Housni. Automatic sign language recognition: A survey. In *2017 International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, page 1–7. IEEE, May 2017. ISBN 978-1-5386-0551-6. doi: 10.1109/ATSIP.2017.8075561. URL http://ieeexplore.ieee.org/document/8075561/.

E. Escobedo-Cardenas and G. Camara-Chavez. A robust gesture recognition using hand local data and skeleton trajectory. In *2015 IEEE International Conference on Image Processing (ICIP)*. IEEE, sep 2015. doi: 10.1109/icip.2015.7350998. URL https://doi.org/10.1109/icip.2015.7350998.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. Online, 2012. URL http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

H.-S. Fang, G. Lu, X. Fang, J. Xie, Y.-W. Tai, and C. Lu. Weakly and semi supervised human body part parsing via pose-guided knowledge transfer. *CVPR*, 2018.

C. F. F. C. Filho, R. S. de Souza, J. R. dos Santos, B. L. dos Santos, and M. G. F. Costa. A fully automatic method for recognizing hand configurations of brazilian sign language. *Research on Biomedical Engineering*, 33(1):78–89, mar 2017. doi: 10.1590/2446-4740.03816. URL https://doi.org/10.1590/2446-4740.03816.

D. A. Forsyth and M. M. Fleck. Body plans. In *Proceedings of IEEE computer society conference on computer vision and pattern recognition*, pages 678–683. IEEE, 1997.

K.-S. Fu and J. Mui. A survey on image segmentation. *Pattern recognition*, 13(1):3–16, 1981.

R. Gandhi. R-cnn, fast r-cnn, faster r-cnn, yolo — object detection algorithms, Jul 2018. URL https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e. [Accessed June 2nd, 2019].

S. Gattupalli, A. Ghaderi, and V. Athitsos. Evaluation of deep learning based pose estimation for sign language recognition. In *Proceedings of the 9th ACM International*

Conference on PErvasive Technologies Related to Assistive Environments - PETRA '16, page 1–7. ACM Press, 2016. ISBN 978-1-4503-4337-4. doi: 10.1145/2910674.2910716. URL http://dl.acm.org/citation.cfm?doid=2910674.2910716.

R. M. Geraci. *Apocalyptic AI: Visions of heaven in robotics, artificial intelligence, and virtual reality*. Oxford University Press, 2012.

R. Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *arXiv:1311.2524 [cs]*, Nov 2013. URL http://arxiv.org/abs/1311.2524. arXiv: 1311.2524.

A. Gonzalez-Garcia, D. Modolo, and V. Ferrari. Objects as context for detecting their semantic parts. *arXiv:1703.09529 [cs]*, Mar 2018. URL http://arxiv.org/abs/1703.09529. arXiv: 1703.09529.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

Z. Gu, Z. Jia, and H. Choset. Adversary a3c for robust reinforcement learning. *International Conference on Learning Representations*, 2018.

Y. Guo. *Deep learning for visual understanding*. PhD thesis, Leiden University, 2017. URL https://openaccess.leidenuniv.nl/bitstream/handle/1887/52990/Thesis_Yanming_Guo.pdf.

Y. Guo, Y. Liu, T. Georgiou, and M. S. Lew. A review of semantic segmentation using deep neural networks. *International Journal of Multimedia Information Retrieval*, 7(2): 87–93, Jun 2018. ISSN 2192-662X. doi: 10.1007/s13735-017-0141-z.

HandTalk. Handtalk. Online, 2019. URL http://www.handtalk.me/. [Accessed May 31st, 2019].

K. He, G. Gkioxari, P. Dollar, and R. Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, page 2961–2969, 2017. URL http://openaccess.thecvf.com/content_iccv_2017/html/He_Mask_R-CNN_ICCV_2017_paper.html.

D. Hebb. The organization of behavior. a neuropsychological theory. *Journal of Clinical Psychology*, 1949.

G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

G. E. Hinton, T. J. Sejnowski, et al. Learning and relearning in boltzmann machines. *Parallel distributed processing: Explorations in the microstructure of cognition*, 1(282-317):2, 1986.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

D. Hogg. Model-based vision: a program to see a walking person. *Image and Vision computing*, 1(1):5–20, 1983.

K. Holstein, J. W. Vaughan, H. Daumé III, M. Dudík, and H. Wallach. Improving fairness in machine learning systems: What do industry practitioners need? *arXiv preprint arXiv:1812.05239*, 2018.

J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.

R. Hu, P. Dollar, K. He, T. Darrell, and R. Girshick. Learning to segment every thing. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, page 4233–4241. IEEE, Jun 2018. ISBN 978-1-5386-6420-9. doi: 10.1109/CVPR.2018.00445. URL https://ieeexplore.ieee.org/document/8578543/.

D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106–154, 1962.

M. Huh, P. Agrawal, and A. A. Efros. What makes imagenet good for transfer learning? *arXiv preprint arXiv:1608.08614*, 2016.

A. Ivakhnenko and V. Lapa. *Cybernetics and forecasting techniques*. Modern analytic and computational methods in science and mathematics. American Elsevier Pub. Co., 1967. URL https://books.google.com.br/books?id=rGFgAAAAMAAJ.

A. G. Ivakhnenko. Polynomial theory of complex systems. *IEEE transactions on Systems, Man, and Cybernetics*, pages 364–378, 1971.

A. Kawamoto, D. Bertolini, and M. Barreto. A dataset for electromyography-based dactylology recognition. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, oct 2018. doi: 10.1109/smc.2018.00408. URL https://doi.org/10.1109/smc.2018.00408.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

A. Kuenburg, P. Fellinger, and J. Fellinger. Health care access among deaf people. *The Journal of Deaf Studies and Deaf Education*, 21(1):1–10, Jan 2016. ISSN 1081-4159. doi: 10.1093/deafed/env042.

J. Lapiak. Sign language - asl dictionary | handspeak, 2019. URL https://www.handspeak.com/.

Y. LeCun, Y. Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436, 2015.

S. J. Lee, S. W. Kim, W. Kwon, G. Koo, and J. P. Yun. Selective distillation of weakly annotated gtd for vision-based slab identification system. *IEEE Access*, 7:23177–23186, 2019.

C. Lim. Mask r-cnn, 2017. URL https://www.slideshare.net/windmdk/mask-rcnn.

T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection, 2016.

Z. C. Lipton. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*, 2016.

P. Lison. An introduction to machine learning. Lecture Notes, 2012. URL http://folk.uio.no/plison/pdfs/talks/machinelearning.pdf.

J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

M. Long, H. Zhu, J. Wang, and M. I. Jordan. Deep transfer learning with joint adaptation networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2208–2217. JMLR. org, 2017.

F.-J. H. Marc'Aurelio Ranzato, Y.-L. Boureau, and Y. LeCun. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

D. Mazzini, M. Buzzelli, D. P. Pauy, and R. Schettini. A cnn architecture for efficient semantic segmentation of street scenes. In *2018 IEEE 8th International Conference* *on Consumer Electronics - Berlin (ICCE-Berlin)*, page 1–6, Sep 2018. doi: 10.1109/ICCE-Berlin.2018.8576193.

W. McCullough and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of mathematical biophysics*, 5(4):115–127, 1943.

A. S. Mehravari, K. Emmorey, C. S. Prat, L. Klarman, and L. Osterhout. Brain-based individual difference measures of reading skill in deaf and hearing adults. *Neuropsychologia*, 101:153–168, Jul 2017. ISSN 0028-3932. doi: 10.1016/j.neuropsychologia.2017.05.004.

M. Mendes de Assis. Aplicação de técnicas de inteligência computacional para reconhecimento de sinais de libras. Bachelor's thesis, UFMG, 2018.

M. Minsky and S. Papert. Perceptron: an introduction to computational geometry. *The MIT Press, Cambridge, expanded edition*, 19(88):2, 1969.

M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. Muharemagic. Deep learning applications and challenges in big data analytics. *Journal of Big Data*, 2(1):1, 2015.

M. A. Nielsen. *Neural Networks and Deep Learning*. Determination Press, 2015. URL http://neuralnetworksanddeeplearning.com.

S. Obdržálek, G. Kurillo, J. Han, T. Abresch, R. Bajcsy, et al. Real-time human pose detection and tracking for tele-rehabilitation in virtual reality. *Studies in health technology and informatics*, 173:320–324, 2012.

R. Parmar. Detection and segmentation through convnets, Sep 2018. URL https://towardsdatascience.com/detection-and-segmentation-through-convnets-47aa42de27ea.

J. Patterson and A. Gibson. *Deep learning: A practitioner's approach*. " O'Reilly Media, Inc.", 2017.

T. Pfister. *Advancing Human Pose and Gesture Recognition*. PhD thesis, University of Oxford, 2015.

W. Rawat and Z. Wang. Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9):2352–2449, 2017.

S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

T. M. Rezende. Aplicação de técnicas de inteligência computacional para análise da expressão facial em reconhecimento de sinais de libras. Master's thesis, UFMG, 2016. URL https://www.ppgee.ufmg.br/defesas/1393M.PDF.

A. Rosebrock. Keras mask r-cnn. Online, Jun 2019. URL https://www.pyimagesearch.com/2019/06/10/keras-mask-r-cnn/. [Accessed June 10th, 2019].

F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

W. Sandler and D. C. Lillo-Martin. *Sign language and linguistic universals*. Cambridge University Press, 2006. ISBN 978-0-521-48248-6.

T. J. Sejnowski. *The deep learning revolution*. MIT Press, 2018.

R. Shokri and V. Shmatikov. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 1310–1321. ACM, 2015.

H.-I. Suk, S.-W. Lee, D. Shen, A. D. N. Initiative, et al. Hierarchical feature representation and multimodal fusion with deep learning for ad/mci diagnosis. *NeuroImage*, 101: 569–582, 2014.

S. Sun, C.-F. Yeh, M. Ostendorf, M.-Y. Hwang, and L. Xie. Training augmentation with adversarial examples for robust speech recognition. *arXiv preprint arXiv:1806.02782*, 2018.

T. Takikawa, D. Acuna, V. Jampani, and S. Fidler. Gated-scnn: Gated shape cnns for semantic segmentation. *CoRR*, abs/1907.05740, 2019. URL http://arxiv.org/abs/1907.05740.

A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660, 2014.

D. S. Touretzky and G. E. Hinton. Symbols among the neurons: Details of a connectionist inference architecture. In *IJCAI*, volume 85, pages 238–243, 1985.

J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.

VLibras. Vlibras - tradução de português pra libras. Online, 2019. URL http://www.vlibras.gov.br/. [Accessed May 31st, 2019].

H. Wang, F. Zhang, X. Xie, and M. Guo. Dkn: Deep knowledge-aware network for news recommendation. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 1835–1844. International World Wide Web Conferences Steering Committee, 2018a.

T. Wang, Y. Li, J. Hu, A. Khan, L. Liu, C. Li, A. Hashmi, and M. Ran. A survey on vision-based hand gesture recognition. In *International Conference on Smart Multimedia*, pages 219–231. Springer, 2018b.

S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. *arXiv:1602.00134 [cs]*, Jan 2016. URL http://arxiv.org/abs/1602.00134. arXiv: 1602.00134.

K. Weiss, T. M. Khoshgoftaar, and D. Wang. A survey of transfer learning. *Journal of Big Data*, 3(1):9, May 2016. ISSN 2196-1115. doi: 10.1186/s40537-016-0043-6.

WHO. Who | estimates. Online, 2019. URL http://www.who.int/deafness/estimates/en/.

B. Widrow and M. E. Hoff. Adaptive switching circuits. Technical report, Stanford Univ Ca Stanford Electronics Labs, 1960.

N. Wiener. *Cybernetics or Control and Communication in the Animal and the Machine*. Technology Press, 1948.

H. Wu, J. Zhang, K. Huang, K. Liang, and Y. Yizhou. Fastfcn: Rethinking dilated convolution in the backbone for semantic segmentation. In *arXiv preprint arXiv:1903.11816*, 2019.

B. Xiao, H. Wu, and Y. Wei. Simple baselines for human pose estimation and tracking. *arXiv:1804.06208 [cs]*, Apr 2018. URL http://arxiv.org/abs/1804.06208. arXiv: 1804.06208.

T. Yaguchi and M. S. Nixon. Transfer learning based approach for semantic person retrieval. In *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, page 1–6. IEEE, Nov 2018. ISBN 978-1-5386-9294-3. doi: 10.1109/AVSS.2018.8639129. URL https://ieeexplore.ieee.org/document/8639129/.

S. Yang, P. Luo, C. C. Loy, and X. Tang. Wider face: A face detection benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Y. Yang and D. Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE transactions on pattern analysis and machine intelligence*, 35(12):2878–2890, 2012.

G. Zanon, R. R. Guerra, , M. M. Assis, T. M. Rezende, G. T. B. Almeida, S. G. M. Almeida, C. L. Castro, and F. G. Guimarães. Desenvolvimento de uma base de dados de sinais de libras para aprendizado de máquina: estudo de caso com cnn 3d. In *14ª Simpósio Brasileiro de Automação Inteligente*, page TBA. SBA, 2019.

L. Zhang, G. Zhu, P. Shen, J. Song, S. Afaq Shah, and M. Bennamoun. Learning spatiotemporal features using 3dcnn and convolutional lstm for gesture recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3120–3128, 2017.

J. Zhao, J. Li, Y. Cheng, L. Zhou, T. Sim, S. Yan, and J. Feng. Understanding humans in crowded scenes: Deep nested adversarial learning and a new benchmark for multi-human parsing. *arXiv preprint arXiv:1804.03287*, 2018.

B. Zia, R. Illikkal, and B. R. Use transfer learning for efficient deep learning training on intel® xeon® processors. Online, March 2018. URL https://software.intel.com/en-us/articles/use-transfer-learning-for-efficient-dl-training-on-intel-xeon-processors.

C. Zimmermann and T. Brox. Learning to estimate 3d hand pose from single rgb images. Technical report, arXiv:1705.01389, 2017. URL https://lmb.informatik.uni-freiburg.de/projects/hand3d/. https://arxiv.org/abs/1705.01389.

H. Zuo. Implementation of hci software interface based on image identification and segmentation algorithms. In *2016 Online International Conference on Green Engineering and Technologies (IC-GET)*, pages 1–6. IEEE, 2016.