THE REPUBLIC OF TURKEY
BAHCESEHIR UNIVERSITY

3D POSE ESTIMATION FROM STEREO IMAGES

Master's Thesis

YILMAZ CENGIZ AKARSU

ISTANBUL, 2019

THE REPUBLIC OF TURKEY
BAHCESEHIR UNIVERSITY

THE GRADUATE SCHOOL OF NATURAL
AND APPLIED SCIENCES
COMPUTER ENGINEERING

3D POSE ESTIMATION FROM STEREO IMAGES

Master's Thesis

YILMAZ CENGIZ AKARSU

Supervisor: ASSIST. PROF. DR. TARKAN AYDIN

ISTANBUL, 2019

THE REPUBLIC OF TURKEY
BAHCESEHIR UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
COMPUTER ENGINEERING

Name of the thesis: 3D Pose Estimation From Stereo Images
Name/Last Name of the Student: Yılmaz Cengiz AKARSU
Date of the Defense of Thesis: 11.06.2019

The thesis has been approved by the Graduate School of Natural and Applied Sciences

Assist. Prof. Dr. Yücel Batu SALMAN
Graduate School Director

I certify that this thesis meets all the requirements as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Tarkan AYDIN
Program Coordinator

This is to certify that we have read this thesis and we find it fully adequate in scope, quality and content, as a thesis for the degree of Master of Science.

Examining Comittee Members	Signature
Thesis Supervisor Assist. Prof. Dr. Tarkan AYDIN	----------
Member Assist. Prof. Dr. Serkan AYVAZ	----------
Member Assist. Prof. Dr. Mürüvvet Aslı AYDIN	----------

ACKNOWLEDGEMENTS

I would like to thank my family for their patience and never- ending support all my writing thesis period.

Furthermore, I want to thank Asst. Prof. Dr. Tarkan AYDIN for being an amazing supervisor for my thesis. I benefited very much from his knowledge, suggestions, and support during my research period.

June 2019 Yılmaz Cengiz AKARSU

ABSTRACT

3D POSE ESTIMATION FROM STEREO IMAGES

Yılmaz Cengiz AKARSU

Computer Engineering

Thesis Supervisor: Assist. Prof. Dr. Tarkan AYDIN

June 2019, 45 pages

In this study, we aim to estimate 3-dimensional (3D) human poses using stereo images. Our approach uses a Convolutional Neural Network model that creates heat maps and location maps to estimate the location of each joint on 2-dimensional (2D) images to create a human skeletal structure. The 2D coordinates of the human skeleton joints are estimated by the information obtained from the operation of this model. By calculating the 3D joint positions, it is aimed to obtain results that are comparable with the current study's results.

A limited set of 3D pictures, which are taken with a 3D stereo RGB camera in different environments and in different poses such as walking, sitting on a chair, eating, greeting, etc. 21 different human skeleton joint locations were obtained in 2D plane by processing separated left and right images, using the trained convolutional neural network model presented in this study. The depth is calculated by using the stereo matching method from the joint locations found and the 2D stereo images to obtain 3D human skeleton pose. Since the collection of 3D stereo data is costly and there are no publicly available data sets with all the information we need to calculate the depth, a data set containing images in different poses has been created and used for testing. The comparisons made between the referenced study's results and our results.

Keywords: Convolutional Neural Networks, Human Pose Estimation, Deep Learning, Stereo Images.

ÖZET

STEREO GÖRÜNTÜLERDEN 3B POZ TAHMİNİ

Yılmaz Cengiz AKARSU

Bilgisayar Mühendisliği

Tez Danışmanı: Dr. Öğr. Üyesi Tarkan AYDIN

Haziran 2019, 45 sayfa

Bu tez çalışmasında, stereo görüntüden 3 boyutlu (3B) insan pozu tahmin edilmeye çalışılmıştır. Yaklaşımımız, bir insanın iskelet yapısını oluşturmak için 2 boyutlu (2B) görüntüler üzerinde her bir vücut noktasını tahminlemek ve bulmak için ısı haritalarını ve konum haritalarını oluşturan evrişimli sinir ağı modelini kullanmaktadır. Bu modelin çalışmasından elde edilen bilgilerle insan iskeleti eklemlerinin 2B koordinatları tahmin edilmektedir. 3B eklem konumları hesaplanarak, referans alınan güncel çalışma ile karşılaştırılabilecek sonuçlar elde edilmesi hedeflenmiştir.

3B stereo RGB kamera ile çekilmiş olan 3B sınırlı resim seti yürüyüş, sandalye üzerinde oturma, yemek yeme, selamlaşma v.b. gibi farklı çevre pozisyonları ile oluşturulmuştur. Görüntüler, sol ve sağ resim olarak ikiye ayrıldıktan sonra referans alınan güncel çalışmada sunulan, eğitilmiş evrişimsel sinir ağ modeli, her iki resim için ayrı ayrı çalıştırarak 2B düzlemde 21 farklı insan iskelet noktası tahmin edilmiştir. Bulunan noktalardan yola çıkarak derinlik hesaplanmış ve 2B stereo resimden stereo eşleştirme yöntemi ile 3B insan iskeleti pozu elde edilmesi amaçlanmıştır. 3B stereo verinin toplanması maliyetli olması ve hazırda gereken tüm bilgileri ile paylaşılmış olan bir veri seti mevcut olmadığından dolayı az veri ile en iyi karşılaştırma sonuçlarına erişmek için farklı pozisyonlardan görüntüler barındıran bir veri seti oluşturulmuş ve testler bunun üzerinden yapılmıştır. Elde edilen sonuçlar referans alınan çalışma ile karşılaştırılmıştır.

Anahtar Kelimeler: Konvolüsyonel Sinir Ağları, İnsan Poz Tahmini, Derin Öğrenme, Stereo Görüntüler.

CONTENTS

TABLES

FIGURES

ABBREVIATIONS

2D	:	Two Dimensional
3D	:	Three Dimensional
AI	:	Artificial Intelligence
ANNs:	:	Artificial Neural Networks
CNN	:	Convolutional Neural Network
CPU	:	Central Processing Unit
CV	:	Computer Vision
DL	:	Deep Learning
dpi	:	Dots Per Inch
FLIC	:	Frames Labeled in Cinema
FPS	:	Frame Per Second
GPU’s	:	Graphics Processing Units
HPE	:	Human Pose Estimation
MKSDK	:	Microsoft Kinect Software Development Kit
ML	:	Machine Learning
MLP	:	Multi-Layer Perceptron
MoCap	:	Marker-less Motion Capture
mpo	:	Multi Picture Object File
PCP	:	Percentage of Correct Parts
PE	:	Pose Estimation
ReLU	:	Rectified Linear Unit
RGB	:	Red, Blue and Green
SHPED	:	The Stereo Human Pose Estimation Data Set
TPU	:	Tensor Processing Unit

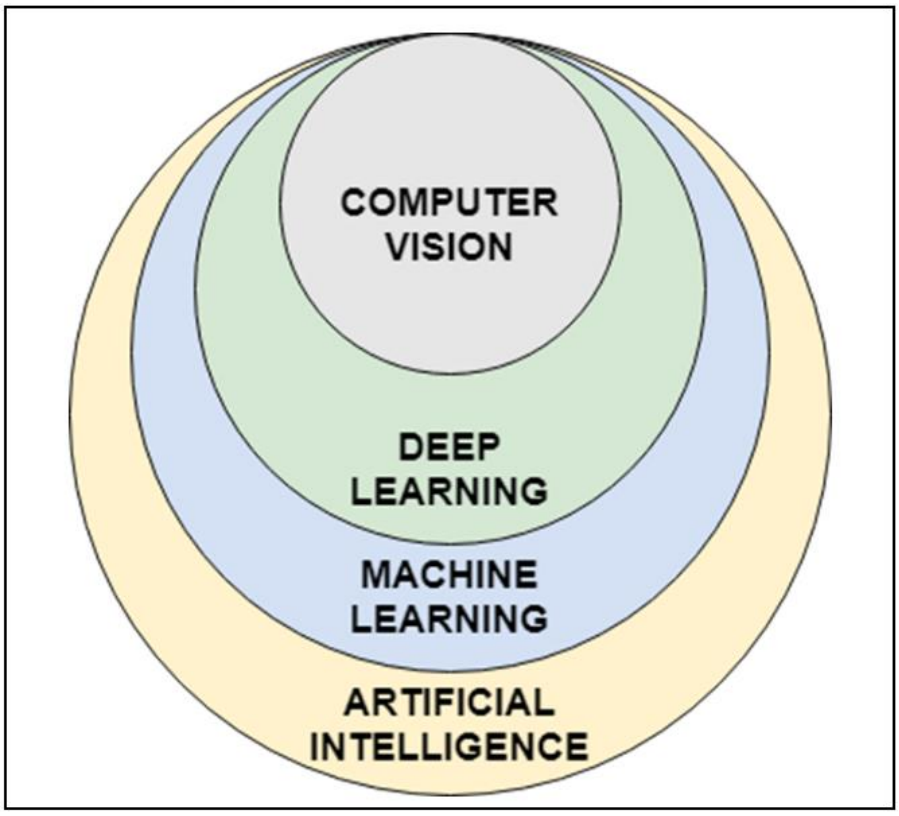
SYMBOLS

millimeter	:	mm
pixels	:	px

1. INTRODUCTION

Acquiring information from a picture or video is an advanced Computer Vision (CV) problem and is examined in detail by many researchers for an effective applicability. Especially in recent years, thanks to the rapid developments in computer hardware, such as central processor units (CPU's), graphics processing units (GPU's), tensor processing unit (TPU), the processing capacity and computational power of computers is far more advanced today. In view of these technological developments, research studies and their practical applications in recent years have been more widely used with the aid of the increasing number of studies on artificial intelligence (AI) and machine learning (ML). In light of these technological developments, AI and ML are increasingly used in many various areas. The relationship between AI, ML, Deep Learning (DL) and CV is showed in Figure 1.1.

Figure 1.1: Overview of The Relationship Between AI, ML, DP and CV



With more general definition, CV is a task of acquiring, processing, analyzing the real-world data on images to produce numerical or symbolic information from it. CV studies accelerated rapidly as a result of advances in machine learning, artificial intelligence and deep learning. Considering the increasing effect of ML, AI and DL in the studies, the studies on computer vision greatly improves and eases the process of the large-scale imaging and video data processing today. Therefore, although CV operations are costly, high time and power consuming; it has a great impact on collecting, analyzing and extracting information from images in many different science applications. This makes it so useful in real-world applications, the number of studies about image acquisition and processing are increasing day by day. Thanks to these researches carried out all over the world, new techniques have enabled the development of CV methods to go further. Each technique brings its own advantages and disadvantages in terms of work or applications and it is effectively used in their own field. One of the areas CV most utilized is image processing and extraction a piece of information from an image which is related to this study scope. The methods are applied in various fields and fields such as medicine, sports, safety, education, virtual reality, etc. (Redlich, 1996), (Besl, 1988)

Using 2D images of objects to obtain a 3D human pose estimation of the object is one of the most important research topics in CV studies. There are different methods to achieve this goal. Particularly, studies on stereo methods are applied in this thesis subject. Even though the stereo method is one of the oldest computer vision methods, it is still being studied extensively by many research groups as a popular problem. One of the main reasons why it is still popular is that it depends on the simplicity in the CV.

Fundamentally, getting results in the light of the information of previous referenced studies of obtained from the 3D full human body pose estimation study with the single RGB camera; this thesis aimed to get results of 3D human full body pose estimation coordinates which can handle jointly person segmentations detection from 2D image joint positions which calculated by pre-trained method (Mehta, 2017) with a 3D RGB camera on a created stereo data set instead of predicts poses

from root-relative joint locations by convolutional neural network (ConvNets or CNNs) pose estimation method.

In this thesis, we focus on finding 3D human pose estimations (HPE) from joint locations on 2D images which predicted by pre-trained CNN from referenced study, called as “Vnect”, (Mehta, 2017). Since collecting 3D stereo human full body pose images is costly and due to lack of 3D data set, a new, limited but 3D data set which contains various scenes is created by us and used in this project.

Our study aims that finding 3D human skeleton joint positions with more accurate rate than the referenced study’s result. For achieving this aim, our contribution is getting 3D joints from 2D by the stereo matching method which 2D joints is estimated by pre-trained CNN model. We added stereo matching method to find 3D skeleton joints to an existing solution for our stereo 3D image dataset which is created by us. As a result of this, our contribution is combining the pre-trained CNN model and stereo matching models to a new solution finding 3D skeleton joint positions.

The referenced previous studies obtain a 3D full human body pose estimation from 2D human joint locations by,

a) Applying bounding box technique on the images,
b) Finding heat map and location maps x, y, z for all joints using CNN regression,
c) Predicting both 2D and root (pelvis) relative 3D joint positions. These operations are costly in term of computing resources. Our study seeks to answer the problem of 3D HPE joint positions by using a computer vision technique called stereo matching (Xu, 2011).

1.1 THESIS STRUCTURE

This study consists of four main parts. The structure of the study is shown below:

a. The first chapter is the literature review chapter. In this chapter, development process and history of the studies related to this thesis will be explained.
b. The second chapter will be about deep learning and computer vision architectures. In this section, the terminology related to the technical structure used in the background of this thesis is defined.
c. The third chapter is named the Methodology and Data. The chapter is the heart of the research proposal. It gives a general overview of using stereo human data set which is created. Then, it gives some explanations on available methods in this field and explains the algorithms used in this study.
d. The fourth chapter is the discussion and conclusion. In this chapter, the proposed method which is evaluated is discussed and make conclusions on that.

2. LITERATURE REVIEW

2.1 INTRODUCTION

Human pose estimation of an image or set of images is estimating spatial positions of human body parts on the target images. The result of that process can be a utility task for many different areas especially based on tracking the human pose states. While most of the state-of-the-art studies focus on 3D human poses from monocular images, lifting from the 2D pose in the image into 3D pose estimations, in recent years, 3D cameras have recently increased their use as they became more affordable. As a result, the number of researches and studies on this subject have increased rapidly.

In this chapter, we will talk about a detailed state-of-the-art literature review of the research which is conducted estimating 3D HPE. The subject of this thesis is to produce 3D skeleton joint position results by applying a different method in addition to current studies, which are getting 3D skeleton HPE from RGB images or videos that are in 2D. In addition, general information and their detail definitions are given which is used in this thesis background studies.

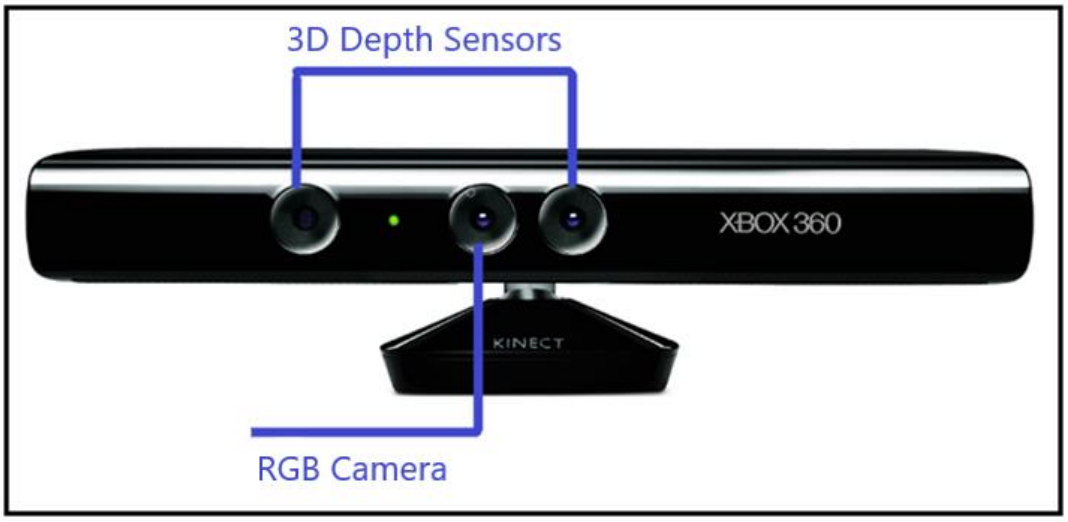
2.2 LITERATURE SURVEY ON 3D HUMAN POSE ESTIMATIONS

Capturing skeletal motion of humans in 3D is widely used in various applications and it has a vast range of applications such as virtual reality, activity recognition, movies, biomechanics, sports video analytics, and autonomous vehicles, human-computer interaction. There are many studies as an attempt to solve and optimize getting 3D pose estimations from an image. In this part, some noteworthy researches will be summarized. From the general to the specific, we will look at the most striking works, which are related with our study. To begin with the date of this study, in recent years depth-based devices (e.g. Microsoft Kinect) and other similar devices played a very important role and have led to many new architectures, studies, and algorithms that address and develop the 3D human body pose

estimation problem (Lo´pez-Quintero, 2017). Depth camera, which sees in 3D, creates a skeleton image of a human and a motion sensor detects their movements in real time. While each study offers a different perspective, these studies also shed light on future studies. In all studies camera calibration is a very critical key point to get more accurate results. Camera calibration is a required stage in computer vision to extract metric information from 2D images converting to 3D (Zhang, 2000).

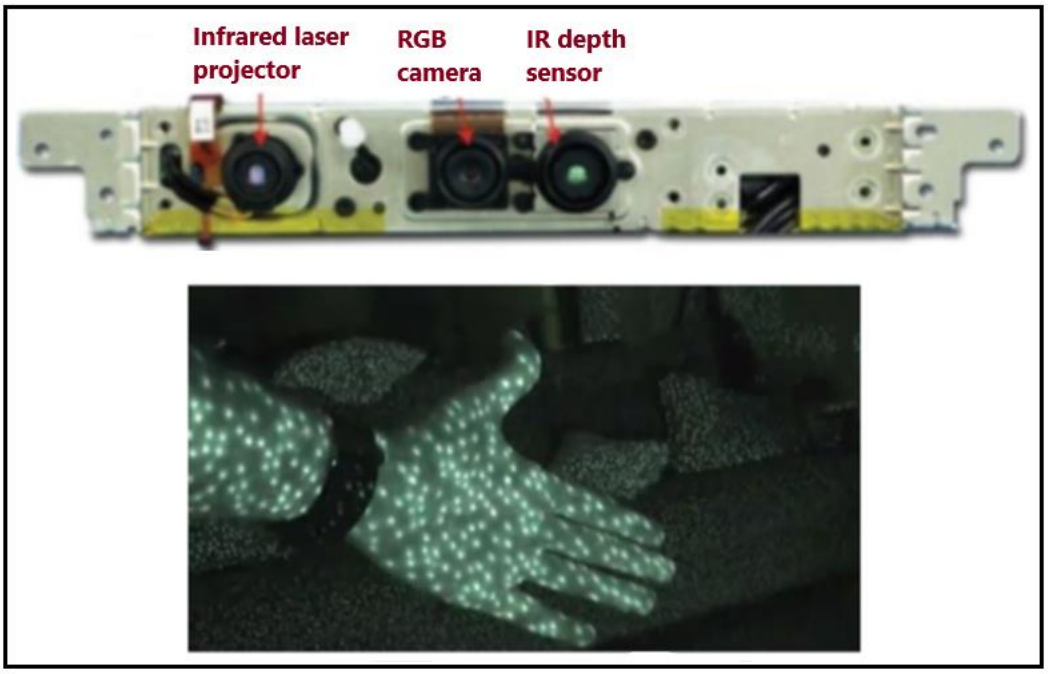
Microsoft Kinect is basically a hardware with a collection of sensors which consists of two infrared projectors and an RGB camera which provide full-body 3D motion capture property (Zeng, 2012), (Tong, 2012). It is a real-time computer-human interaction tool with the frame rate is 30 frame per second (FPS) and it has a stepper motor and mic array to take sound input as well.

Figure 2.1: Microsoft Kinect



Source: (Zeng, 2012)

Figure 2.2: Microsoft Kinect Structure and Signals on a Hand



Source: Graphics.stanford.edu

When using the depth information provided by the Microsoft Kinect camera, each pixel correlates with the estimated distance between the Microsoft Kinect camera and the closest neighboring object in the scene, this information allows the viewing of different parts of the human with the help of the Microsoft Kinect camera and sensors.

To form the Microsoft Kinect human skeleton in real time, the acquired information of the person facing front the camera, processed in various ways. It performs the operations with the Microsoft Kinect Software Development Kit (MKSDK), that is a built-in software package which is supported by Microsoft Kinect camera and does not require pose calibration. Using MKSDK the predetermined 20 joint positions can be predicted (Shingade & Ghotkar, 2014). Since 3D HPE from a single image is a very complex and difficult task because of the lack of missing depth knowledge, depth sensors have been utilized for calculating the depth of 3D HPE (Baak, 2011). However, the depth cameras on devices have a limitation on indoor and outdoor environments because of different light positions and light

density (Yasin, 2016). Studies have focused on overcome these shortcomings on the constraints of RGB-D camera systems. So, to capture the 3D full-body skeleton pose of a human consistent posture using a single cheap RGB camera is a further study that has a history in computer vision (Ferrari, 2009), (Lo´pez-Quintero, 2017).

(Wei, 2016) calculate human skeletal pose estimation in 2D with a single RGB camera which is independent of person shape differences, background variety and distortion due to lighting perspective. The study shows that 2D HPE is stable and calculated successfully. However, these methods only calculate 2D skeletal information Felzenszwalb, (2010), Felzenszwalb and Huttenlocher, (2005), Ferrari, (2009), Wei, (2017). In addition, these studies aimed to demonstrate their performance in real-time Cao, (2016). Predicting 3D pose directly from 2D RGB images has been demonstrated using offline by Tekin, (2016), Bogo, (2016), Nie, Wei, Zhu., (2017). These studies predict the depth of human body parts joint positions separately and estimates 3D pose and approximate human shape from 2D images. Mehta, (2016) created a method for calculating 3D human full body pose estimations from monocular images. Generally, these all studies have an offline working background to find 3D joint positions separately per images in each time of the image sequences. Variety of the per image the architecture of the method often reconstructs 3D human pose joint positions that cause unstable, and it is not give a guarantee for constant bone lengths Zhou, (2015), Mahendran, (2018).

The diversification and proliferation of the work on the technical side of this subject, the rapid improvements artificial intelligence and deep learning studies in the past few decades because of the advance of technology, the development of studies and methods has accelerated. Today the studies focus on a single RGB camera to get the same and even better results from previous studies especially in the wild areas (Mehta, 2016) without the need for studios, any markers stick the human body for estimating joint positions and a device like Microsoft Kinect which is equipped with a sensor**.**

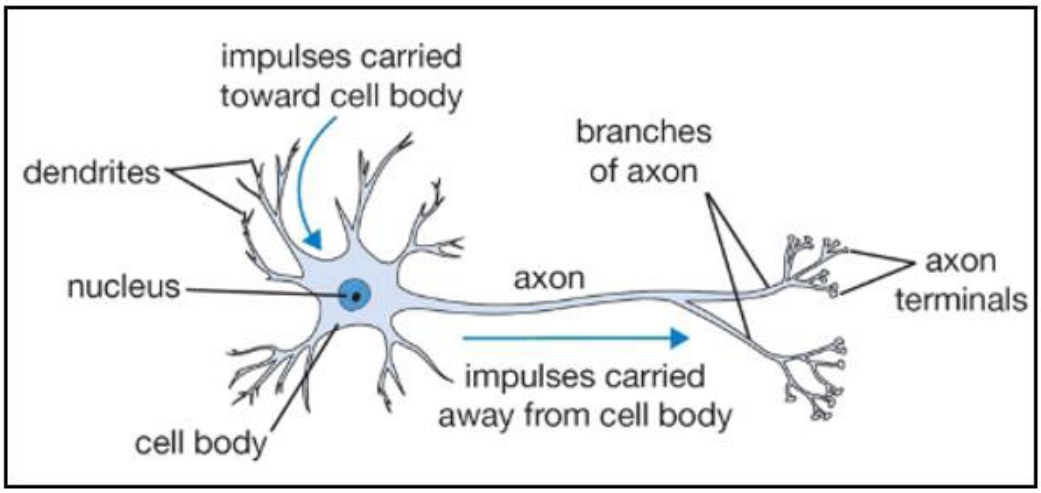
3. DEEP LEARNING AND COMPUTER VISION ARCHITECTURES

In this chapter, general technical content related to the study is discussed. This chapter is divided into four sub-sections, which are artificial neural networks and deep learning, convolutional neural networks and pose estimation in computer vision. In each part, we will specify general definitions, researches and their solutions that addressed the problem of 3D HPE with an RGB camera in general. This thesis study uses CNN to estimate 2D human skeleton joint positions as described details before in the literature review section and obtain the 3D human body pose from this information by the help of the CV stereo matching method. Therefore, in this section definitions are explained related to CNN architecture and pose estimation in CV explained along with the artificial neural networks and deep learning.

3.1 ARTIFICIAL NEURAL NETWORKS AND DEEP LEARNING

The artificial neural networks (ANNs) has an old back history which is starting from the 1940s when a simple model was designed by neurophysiologist and mathematician Warren McCulloch, using electrical circuits to explain how neurons can work Walter Pitts. Basically, it was inspired by the biological neural connection in the human brain. The neural network consists of neural neurons called artificial neurons. Neuronal connections of the human brain composed of connecting electrical pulse transmitting neurons. The fundamental definition neuron is the brain cell. Neurons communicate with each other by propagating electrical impulses through connections called synapses. It gathers the receiving input signals from other cells then makes an output after performing operations. The bonding of neurons in the brain consists of a series of connected units called artificial neurons. A Biological neuron structure is showed in Figure 3.1.

Figure 3.1: A diagram of the human's brain biological neuron model.

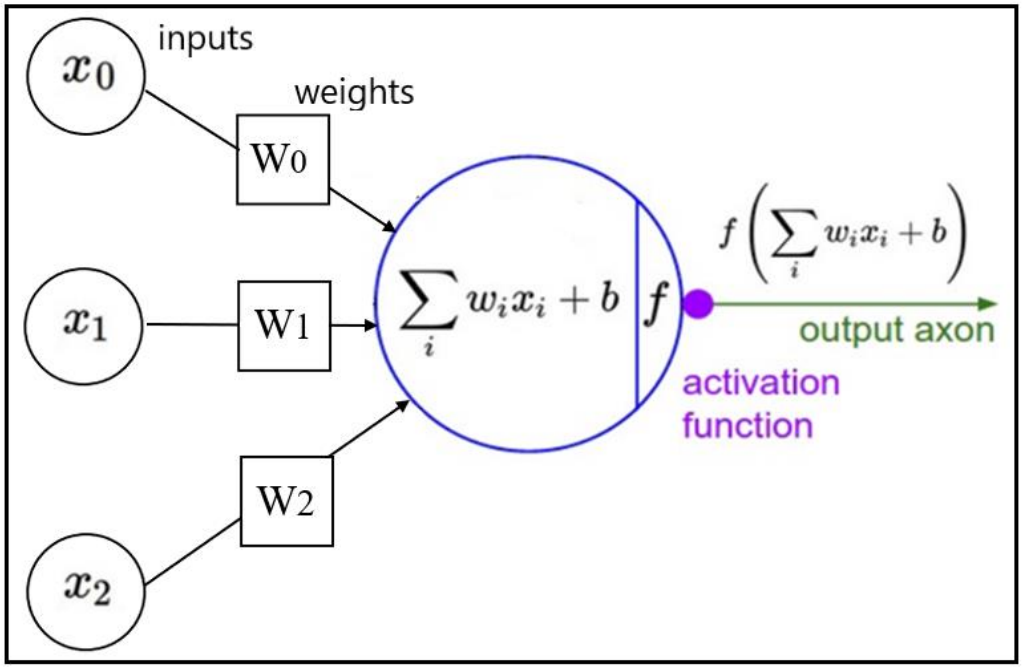


Source: Karpathy Andrej. Stanford Convolutional Neural Networks for Visual Recognition class

As shown in Figure 3.1, the neurons consist of four main parts: Dendrites, Cell Body, Axon and Axon Terminals (Synapses). Dendrites are responsible of the communication channels for transmitting electrical impulses to the cell nucleus. Nucleus is the processing center of the signals. In a biological nervous system, neurons communicate with each other by propagating electrical impulses through connections called synapses. Inspired by these studies and information over the years, Rosenblatt, a psychologist, proposed "the perceptron" concept in the first time in 1958 which is a mathematical model created by inspiration from biological neurons in the human brain.

Rosenblatt proposed a simple rule to compute the output. He introduced weights, w1, w2, ..., real numbers expressing the importance of the respective inputs to the output. The perceptron is composed of a network of units, which are analogous to biological neurons. A unit can receive input from other units. On doing so, it takes the sum of all values, weighted sum, received and decides whether it is going to forward a signal on to other units to which it is connected. This is called activation. The activation function uses some means or other to reduce the sum of input values to a 1 or a 0 (or a value very close to a 1 or 0). The perceptron model is showed in Figure 3.2.

Figure 3.2: A diagram of a perceptron neuron model.

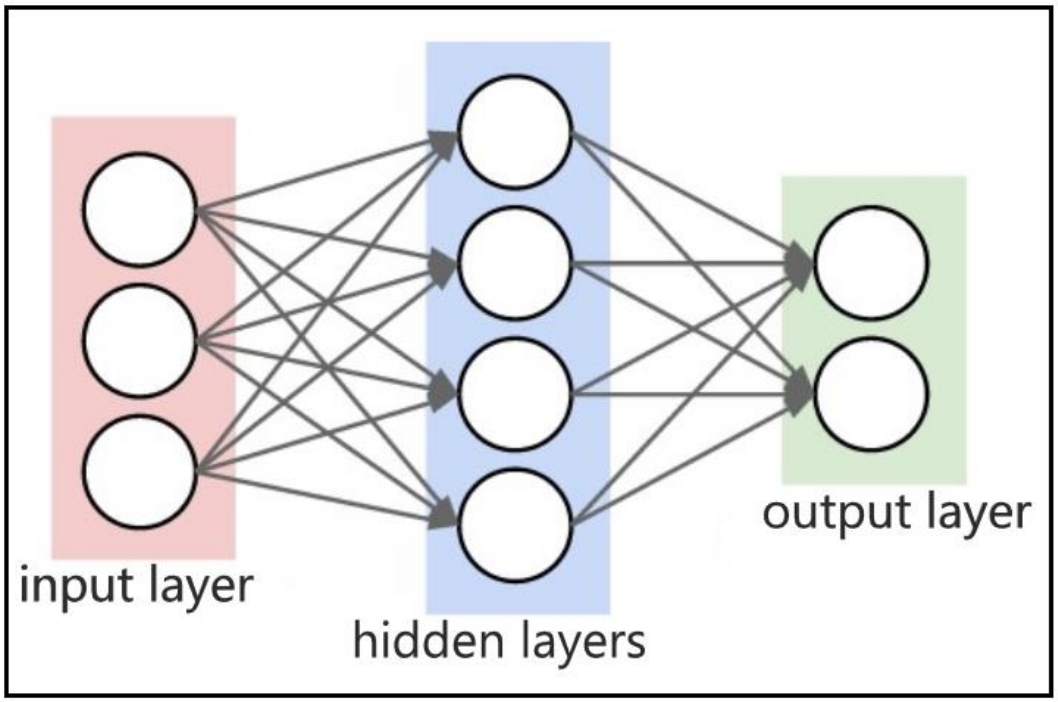


A perceptron takes several binary inputs, x1, x2, ..., and produces a single binary output as a result of some processes. Since, the computing capacity of a perceptron is restricted to a simple logical operation at a time such as “and”, “or”, “not”, it is required to be used more than one perception at the same time for complex mathematical operations to get the output. Therefore, a robust architecture called “layer” has been established to connecting many perceptron with weights in the parallel process. However, the single-unit (or neuron) perceptron can only learn tasks that can be separated linearly. It has been proved by (Minsky and Papert 1969) that it is impossible for a single layer perceptron to learn a simple “XOR” (exclusive or) function. In the ANN architecture, connections between these layers together with the number of layers used are important. Hence, an ANN architecture consists of three parts: The input layer, the output layer and, if necessary, layers between these two layers are called “hidden (middle) layers” (Gonzalez, 2008). Multi-Layer Perceptron (MLP) overcomes these limitations with adding hidden layers (Tinchcombe, 1989). By adding interlayers called hidden layers, these studies continued with created a multi-layered neural network in 1975. In the 1982 Reilly and Cooper, with the help of the multiple layers called hybrid network, each layer solves a separate problem in the network. After that, the backpropagation algorithm

was the first formulated by (Werbos, 1982) to train multi-layer neural networks. After years, with the importance of (Rumelhart, 1986)'s work accepted much more by society. Thus, studies have gained speed again after about 15 years. The period from XOR study to backpropagation is called dark age “AI winter”. However, in those days as the number of layers increases which are more than 2 layers researches having trouble to see the successful results. Because when the network is getting complex and increasing the number of layers requires much more computational power, and that kind of large processing power is not available at the time as a first problem. In those years, researches developed neural networks architecture to solve research problems in their studies, but they were relatively slow. Because of the limitations of hardware, neural networks take weeks to learn that means neural network works as theoretically but insufficient in practical. So, in the light of these studies a neural network consists of three elements which is shown in Figure 3.3.

a) *Input Layer:* This layer accepts incoming data from the outside world to the network, fundamentally there is no computation is performed at this layer, nodes transmit the information to the hidden layer.
b) *Hidden Layer:* In this layer, nodes have the responsibility of performs all kinds of computation on the features that are brought up through the previous layer and transfer the results of the computations to the output layer.
c) *Output Layer:* This layer brings up the information learned by the network to the outer world.

Figure 3.3: Two-layer neural network with three inputs.



Source: Karpathy Andrej. Stanford Convolutional Neural Networks for Visual Recognition class

Another main problem is that there are very few large and usable data sets. Training the network with a restricted data set causes the network to overfit. Gradient based methods learn a parameter's value by understanding how a small change in the parameter's value affects the network's overall result. Also, one of the important problems in deep networks is the vanishing gradients problem. Since it has an extremely small value, it cannot make a significant change to the output of the network. The network cannot learn the parameters effectively. As more layers are added to the structure of the neural network, to update the weights is getting harder because the error signal becomes so small or the signal becomes so approximatively small that the signal registers as 0, making the network hard to train. The simplest solution is to use an activation function, like Sigmoid, Rectified Linear Unit (ReLU), etc. These are the most commonly used activation functions and don't cause a small derivative. To give a general overview, we will only mention these two.

Sigmoid (logistic) function is plotted as 'S' shaped graph above this equation (1):

$$hSigmoid\ (x) = \frac{1}{1+e-x} \quad (1)$$

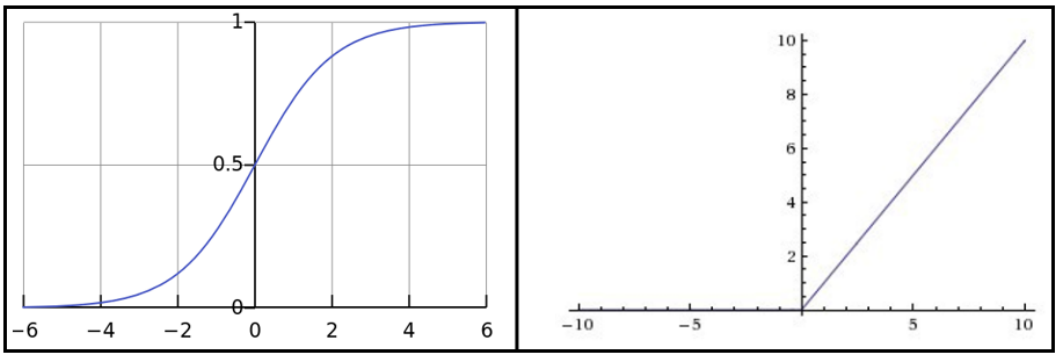
is also showed in Figure 3.4. It is a non-linear function. Combinations of this function are also nonlinear. So, we can stack layers. Usually used in output layer of a binary classification, where result is either 0 or 1, as value for sigmoid function lies between 0 and 1 only so, result can be predicted easily than linear function which can be goes to infinity.

Another activation function is ReLU. This formula (2) (Hinton, 2017)

$$hReLU\ (x) = \max(0, x) \quad (2)$$

which is shown in Figure 3.4. It gives an output x if x is positive and 0 otherwise. This means fewer neurons are sparse activation and the network is lighter.

Figure 3.4: Representation of Sigmoid and ReLU functions.



In 2006, researchers have used a greater number of layers to train in their studies. Neural networks are rebranding as “Deep Learning” by the researchers for more than two hidden layers of architecture in their neural network. (Hinton, 2006) introduced a new idea for a weight initialization method to making a faster algorithm, called greedy algorithm. The idea was to train a simple two-tier unsupervised model, then freeze all the parameters, paste it onto a new layer, and

simply train the parameters for the new layer. The greedy algorithm is initializing on fine-tuning weights using complementary prior studies (Hinton, 1995) and it is a learning algorithm for constructing a layered network, which is one layer at a time. Learning algorithm means even in a deep neural network that has millions of parameters and many hidden layers, the algorithm can find good parameters quickly. This technique is used to set the network initial weights instead of random initialization. So, deep neural networks can be successfully trained with better weight initialization.

To sum up, with the aid of unsupervised pre-training models, the number of studies in this area is increasing tremendously. The promise of deep learning is not that computers will start to think like humans. It indicates that given enough quality data set, fast enough processors, and an advanced enough algorithm, computers can begin to accomplish tasks almost as good as a human.

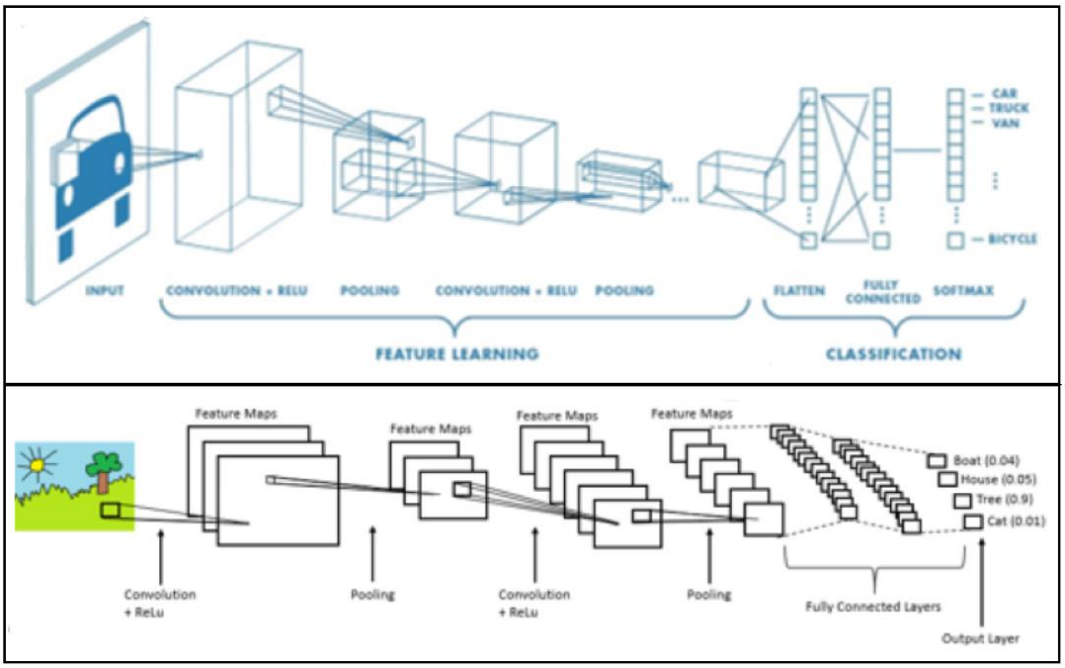
3.2 CONVOLUTIONAL NEURAL NETWORKS

Convolutional Neural Networks (CNN) is a deep artificial neural network which is a special case of the multilayer feed-forward neural networks (LeCun, 1989), (Sichkar, 2018). CNN has changed the way to perform image classification and object recognition. It is comprised of more convolutional layers than MLP and it is designed to enhance in image processing, video, audio and natural language processing (Farabet, 2013), (Krizhevsky, 2012), (LeCun, 2015), (Schmidhuber, 2015). Although, CNN idea was presented in the 1960s, it did not gain popularity as much as today because of some constraints on that day such as technological, cumulative and quality data problem, not many researches have been done on this area, at that time.

Deep neural network is a parallel algorithm. So, more than one algorithm instance runs at the same time. This parallelism is an advantage for thousands of core GPUs to reduce the processing time needed by deep learning networks (Karpathy, 2014). Nowadays, thanks to improving theorical methods, technological improvements, CNN can be used in many different areas. In general CNN uses convolution,

nonlinearity and pooling methods and consists of several steps as is showed in Figure 3.5.

Figure 3.5: The general structure of a convolutional neural network

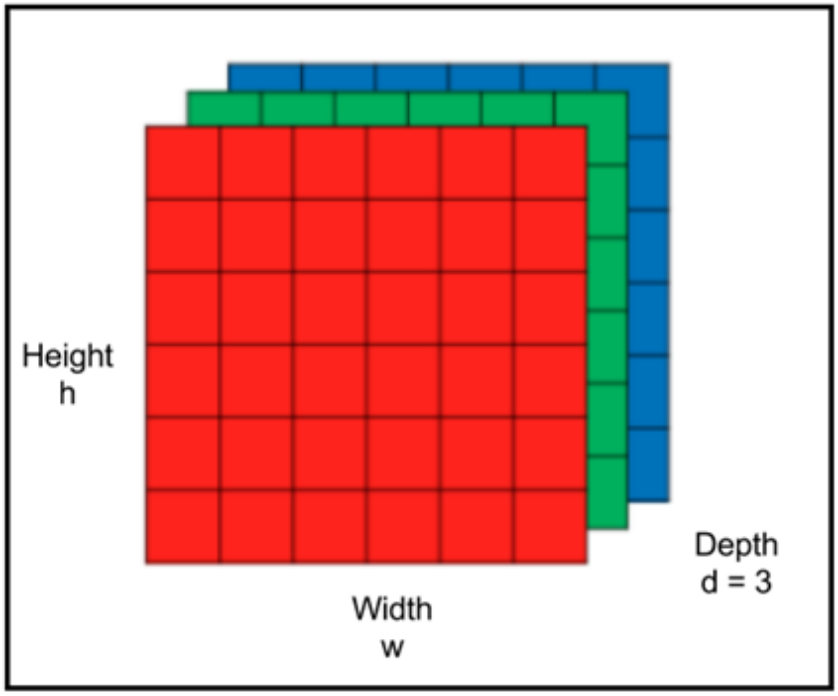


Source: MathWorks - Convolutional Neural Network Notes

Input Layer: Input layer is a real-world, 3D, data that comes from outside of the network. In this example, the input data consists of pixel values in the image RGB channel.

Convolutional Layer: The convolutional layer is an important part of the CNN architecture. The main purpose of the convolutional layer process is extracting some features from the input data. For colored images, the image matrix has a depth of three layers, one for each of the three-color channels: RGB (R)ed, G(reen), B(lue) is showed in Figure 3.6.

Figure 3.6: An RGB image matrix structure



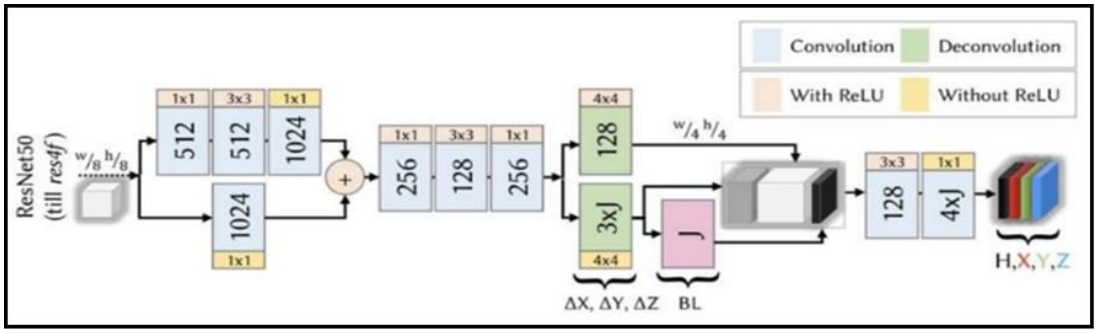
For example, if input data is a car image, then the network would process the image to extract distinguishable features like car's edges, color, physical parts, etc. As seen in Figure 3.2, after convolution step there are some filters can be applied to construct feature map.

Pooling Layer: Pooling layer is another important component of CNN. In this layer reduces the dimension of the feature map which is derived from the previous stage. So, the benefit of this layer is the increase in performance by reducing training time and memory utilization.

Fully Connected Layer: The fully connected layer is the last learning layer in CNN architecture which converts extracted feature maps into output. It outputs a one-dimensional array whose size equal to the number of classes.

In this thesis, CNN used for especially for the task of distinguishing human skeleton joint positions estimation to get a full human body pose estimation. Our study uses the-referenced study's CNN model is shown below:

Figure 3.7: The referenced-study's CNN model architecture.



Source: (Mehta, 2017) – Network structure of CNN model diagram

The structure of Figure 3.7 is preceded by ResNet50. The network predicts 2D locations heat maps H and 3D joint locations $X\,Y,Z$. In addition, the network predicts kinematic root relative locations maps $\Delta X, \Delta Y, \Delta Z$ and it computes bone length as shown in formula:

$$BL = \sqrt{\Delta X\ \Theta\ \Delta Y\ \Theta\ \Delta Z} \qquad (3)$$

3.3 POSE ESTIMATION IN COMPUTER VISION

Pose estimation (PE) means that taking an input photo or video of people and calculate the output which is a description of heir poses. Human pose estimation (HPE) is a highly active research study that recognizes specific elements in the image and finds its position on the coordinate system as pixels in the image of each element (Shapiro & Stockman 2001). In addition, 3D HPE is a growing research topic in the last 20 years (Sigal, 2011). Estimating the spatial coordinates of human body joint positions in a given single, typically monocular image, is a popular (Andriluka, 2012) and a complex task. After using CNN in CV studies, there has been huge progress about HPE in the CV (Szegedy, 2014), (LeCun, 2015), (Tzimiropoulos, 2016). The pose of an object is a combination of position and orientation in a monocular, a stereo or an image sequence.

Nowadays, PE has a significant influence on many applications that can benefit from such technology that especially when the topic is human-computer interaction. In addition, marker-less motion capture (MoCap) technology developed rapidly

over the last years and this technology is still used in many different areas such as games, films, character animation, surveillance, entertainment, and medical purposes. HPE can be classified into three parts (Zhang, 2017).

a) Generative approaches.
b) Discriminative approaches
c) Hybrid approaches

Generative approaches monitor the images or depth map by pairing them with a 3D body model which has tracking result from the previous frame to reduce the search space of the pose in the current frame. This technique does not require training data, and it is suitable for monitoring any type of pose.

Discriminative approaches directly learn from observations of the pose parameters. These methods can process quick motions well, because they learn quite fast. On the other hand, discriminative methods require a large amount of training data and they obtain poor results when the test data contains poses that the training data does not.

Hybrid approaches reach the best performance, they can work for fast movements and do not severely lose track even when the current pose is not in the training data.

This study aims to increase the success rate using the simple an RGB camera instead of expensive equipment and a suitable studio.

4. METHODOLOGY AND DATA

In this chapter, details about methodology and data related to the study will be given. This chapter is divided into four sub-sections, which are about,

i. Stereo vision and object dept analysis,
ii. Stereo human data set,
iii. Training data set on the trained model
iv. Evaluation and error measure metrics

In each part, we will specify general definitions, researches and their solutions that addressed the problem of 3D HPE with an RGB camera in general

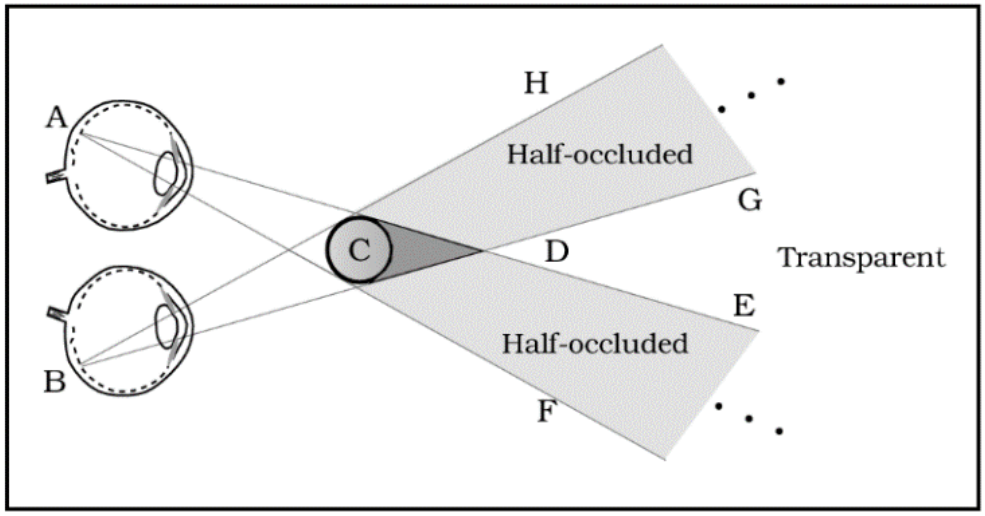
4.1 STEREO VISION AND OBJECT DEPTH ANALYSIS

4.1.1 Stereo Image Modelling

Depth perception is the ability to perceive the world in three dimensions (3D) and to evaluate the distance of objects. Depth perception can be categorized into two parts: Human depth perception and computer dept analysis.

As humans, we can understand the outside world as a real-world 3D scene from both eyes without extra effort (Zisserman, 2003). The main factor that provides this is the two eyes side by side that people have, with an average distance between them. As shown in Figure 4.1, two eyes have intersection zone which is called “C”. In that region, the brain processes different pictures from each eye and combines them to form objects in depth and their 3D shapes, images (Howard & Rogers, 2002). Depth perception makes it possible for the eyes to determine distances between objects.

Figure 4.1: 3D seeing in human which is called stereo vision



Source: foundationsofvision.stanford.edu+

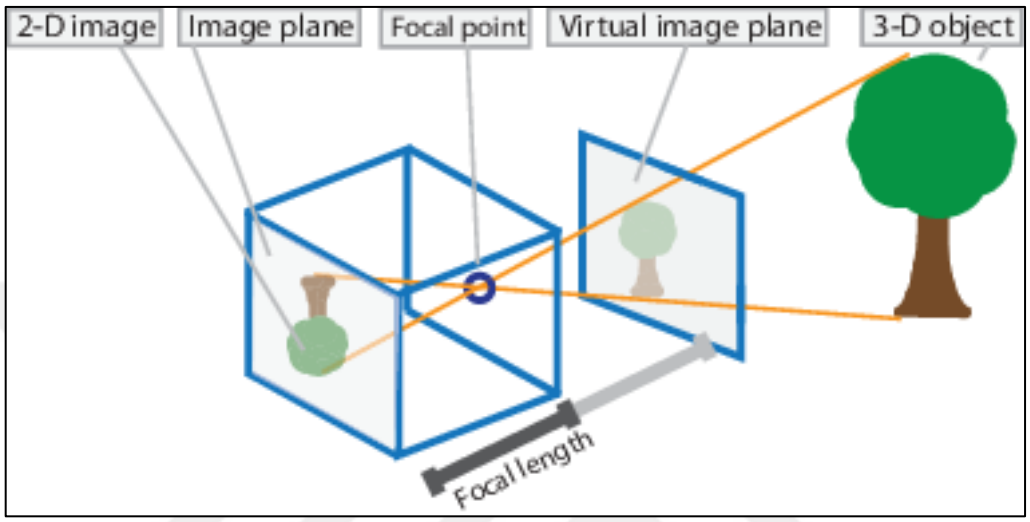
In the computer world, there are two different methods to achieve the depth information and 3D view of a scene by computers: “passive methods” and “active methods”. The first method is aimed at copying the vision system in humans to computers. Although, this method has some difficulties, with the help of artificial intelligence, technological developments and the increasing number of studies, today passive methods are a popular issue and a significant effect on CV. The main purpose of this method is to simulate the detection method in humans for the perception of the real-world as 3D by the CV. The second method is called active methods that have been revealed to overcome the difficulties of the passive method (Surmann, 2007). In addition to the passive methods in active methods, there is a reflector device that sends light to the scene. After the light patterns emitted by this device are reflected on the stage, the depth is calculated with the help of IR depth sensor to try to obtain the 3D vision (Chen, 2011).

4.1.2 Stereo Camera Model

Before defining the stereo camera model, a fundamental model called “pinhole” will be explained. The pinhole is the simplest and widely used camera model, also known as a linear approach model or perspective model. A pinhole camera is a simple camera without a lens and with a single small aperture. Light rays pass

through the aperture and project an inverted image on the opposite side of the camera (Laganière, 2011). Projection of an image by used pinhole model is shown in Figure 4.2.

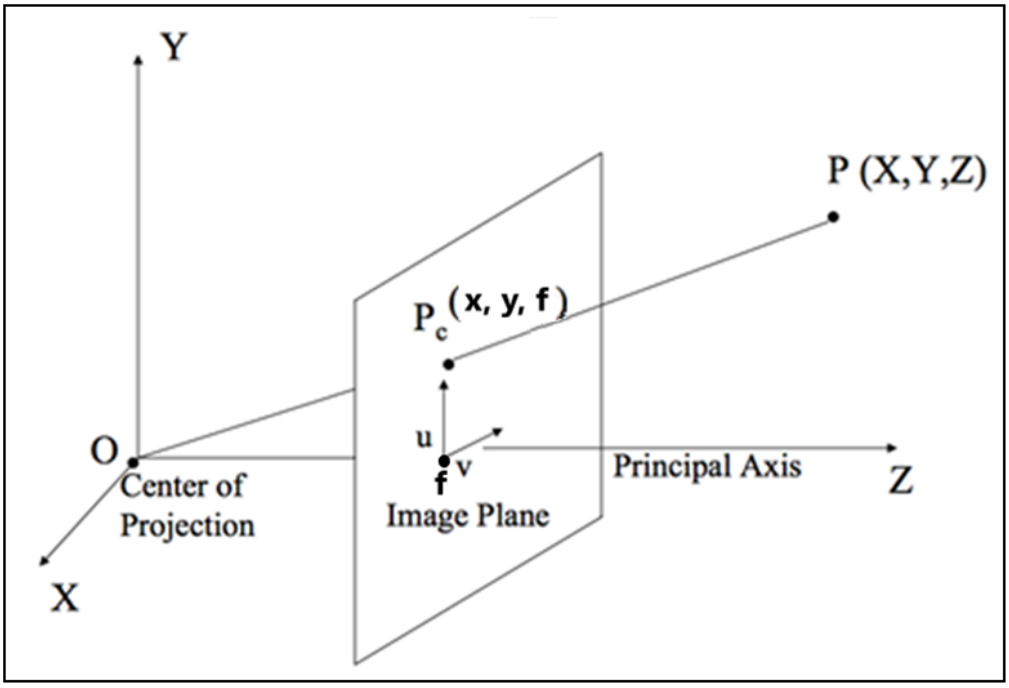
Figure 4.2: The Pinhole Camera Model.



Source: https://uk.mathworks.com/help/vision/ug/camera-calibration.html

As shown in Figure 4.3, the pinhole camera model consists of an image plane placed farther away from the camera center at the **O** point. Where **f** is the focal point of the camera. It is called the optical axis, which leaves the camera center and cuts the image plane vertically and linearly. The point where the image plane and the optical axis intersect is called the image center. In the pinhole model, the image of any **P** point in the real world occurs at the point of point **p** where the right part to be drawn between point **P** and the **O** center of the camera intersects the image plane.

Figure 4.3: The Pinhole Camera Model on Coordinate Plane.



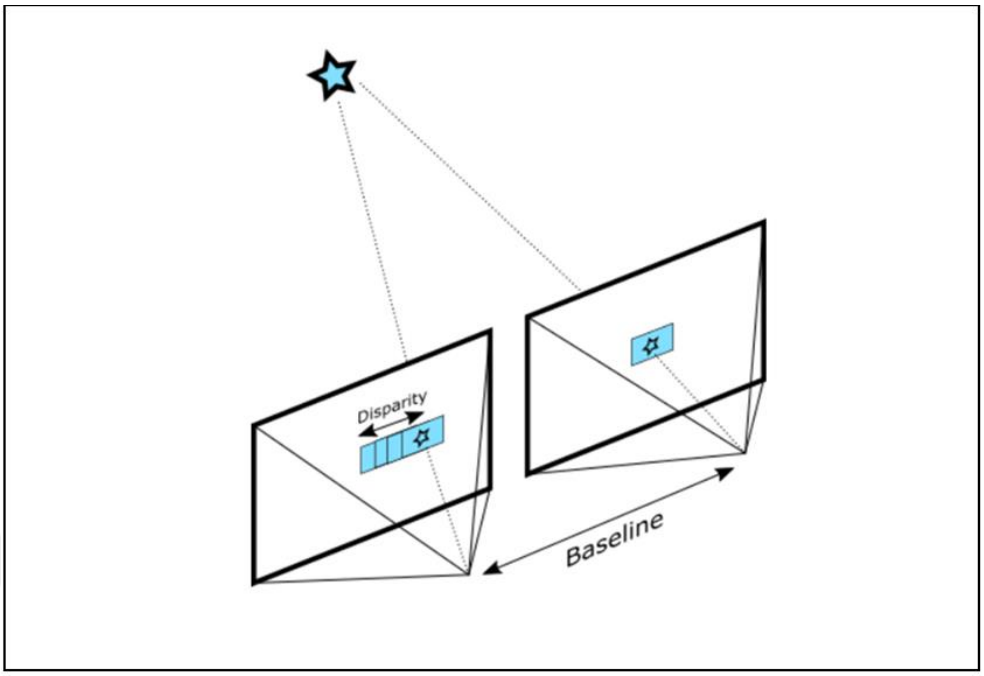
Source: https://prateekvjoshi.com/2014/05/31/understanding-camera-calibration

Relation between $P\ (X, Y, Z)$ and $p\ (x, y, z)$ point on coordinate system is shown in the equation (4):

$$X = f\frac{X}{Z},\ Y = f\frac{Y}{Z},\ Z = f \tag{4}$$

Stereo vision is a technique aims to extract depth information of a scene from multiple-view camera systems. This is done for all pixels of one of the images. Then, the distance between corresponding horizontal pixel locations of points when the two images are superimposed is called disparity map (Mühlmann, 2002). It is shown in Figure 4.4.

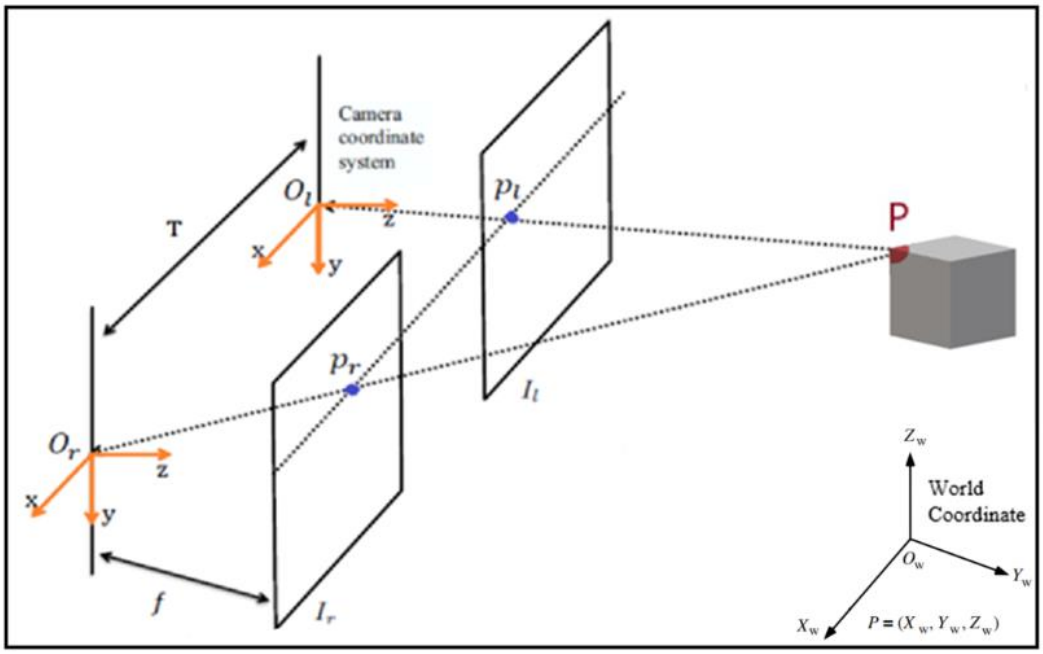
Figure 4.4: Disparity by Matching Blocks in Left and Right Images.



Source: https://raw.githubusercontent.com/wiki/dorodnic/librealsense/stereo-ssd.png

Just the disparity is not enough to calculate the depth value of the 3D point. However, this formula is fair enough to calculate the depth, in this thesis. The results of stereo matching are reliable if the vertical pixel locations are identical between a pair of images. According to the disparity map, differences between the same point's positions on the left and the right image pixel values found are inversely proportional to the distance of objects from the camera plane. Therefore, the first issue to be solved in the stereo problem is to match the image pixels corresponding to the same scene points (Horn, 1986), (Blake & Zisserman, 1987). The Figure 4.5 shows the camera centers on the left and right cameras:

Figure 4.5: Stereo Camera Model.



Il and **Ir** are the image planes of the left and right camera or camera lens. Draw projections to the image planes of any **P** point in the scene and **pl**. The main purpose of the stereo methods is to find **pl** which is shown in the equation (6) and **pr** which is shown in the equation (7) points and to triangulate the depth value of point **P** according to **x** and **y** points on the coordinate system. **f** is focal length and **T** represents baseline of the distance between the camera centers which is shown in the equation (5). There is a relationship between the points in the Figure 3.4.3.3 coordinate system as follows:

$$T = |Ol - Or| \tag{5}$$

$$p_l = \frac{\left(X - \frac{T}{2}\right)f}{Z} \tag{6}$$

$$p_r = \frac{\left(X + \frac{T}{2}\right)f}{Z} \tag{7}$$

From these equations $d = x_l - x_r$ can be calculated. This result is called disparity (d) value (Adrian, 2008).

$$p_l - p_r = \frac{Tf}{Z} \quad (8)$$

When these values are calculated, coordination points of $P\ (X, Y, Z)$ can be found where Z is depth. The equation to find X point is shown in the equation (9), Y point is shown in the equation (10) and the formulation of depth information is shown in (11):

$$X = \frac{T(p_l + p_r)}{2d} \quad (9)$$

$$Y = \frac{Ty}{d} \quad (10)$$

$$Z = \frac{Tf}{d} \quad (11)$$

After finding these camera coordinates, according to the real-world coordinates of the position of objects can be found after the measurement of the camera (Fua, 1993), (Grosky & Tamburino, 1990).

4.2 STEREO HUMAN DATASET

Most of the existing studies use the shared, well-known data sets in their studies such as Frames Labeled in Cinema (FLIC) data set, Human 3.6M, MPII Human Pose Data set, The Stereo Human Pose Estimation Data set (SHPED). (Iqbal, 2017), (Nie, Wei, Zhu, 2017), (Dushyant, 2017)**.** Also, SHPED is contains some stereo images from (YouTube) with the filtering for stereo images keyword "yt3d:enable=true".

Although SHPED has some stereo images, we cannot use the images because our model depends on the 3D camera (FujiFilm Finepix Real 3D W1 camera) which has a known focal and base length. We cannot know the camera properties of the

other data sets even if they may have also stereo images. Our camera has a base length of 77 millimeters (mm) and a focal length of 35 – 105 mm. Focal length is dependent on the digital zoom level applied (FujiFilm). The camera is shown in the Figure 4.6.

Figure 4.6: The camera that is used to create our stereo data set.

Source: https://www.fujifilm.com - Fujifilm, FinePix REAL 3D W1

We create our own data set in various natural environment scene to get the best results when comparing the referenced state-of-the-art study. Fundamentally, our images are different positions on daily life positions (walking, sitting, waiting, doing sport, taking photo, posing, discussion and walking together) which is showed in Figure 4.9.

The well-known data sets usually provided ground-truth data. We created our own novel data set. So, all ground-truth points of 21 skeleton joints are prepared by us. We have 8 different action poses and 3 different images for each pose. For this purpose, we use “Paint” program to find x and y coordinates by hand for each 21 joints which is shown for “head” positions in Figure 4.7. Then, we got 2D coordinate locations for every joint position for every activation poses. for each joint.

For example, the red square shows the position of “head” in Figure 4.7, and “shoulder-right” in Figure 4.8 for our dataset’s “walking” pose. The rest 20 joint positions are found with the same technique. Our aim to prepare our annotated ground-truth locations file.

Figure 4.7: “Head” ground-truth position using “Paint”.

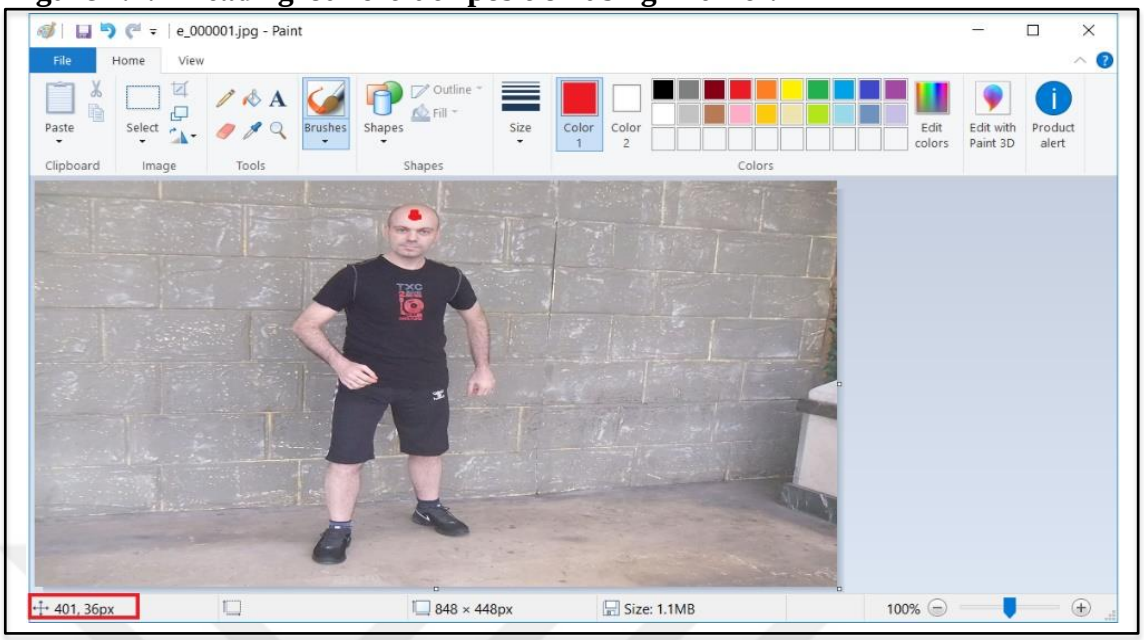


Figure 4.8: “Shoulder-right” ground-truth positions using “Paint”.

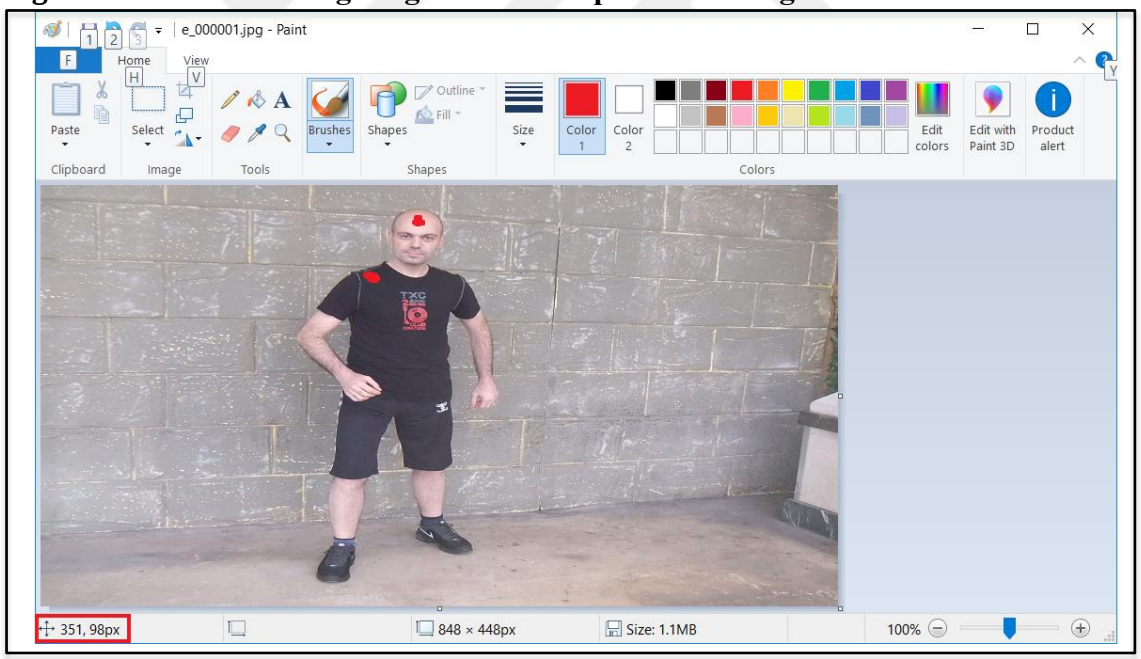
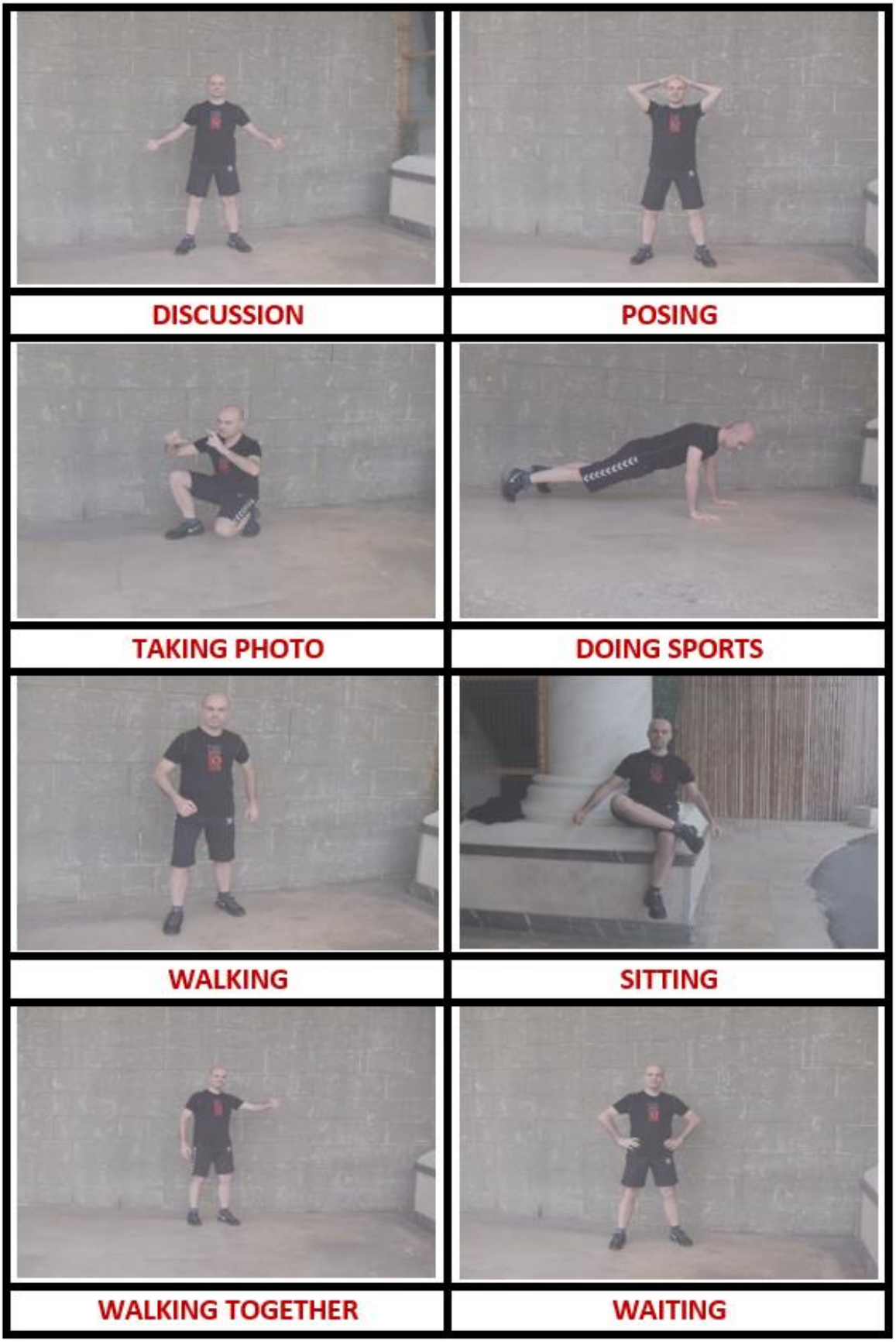


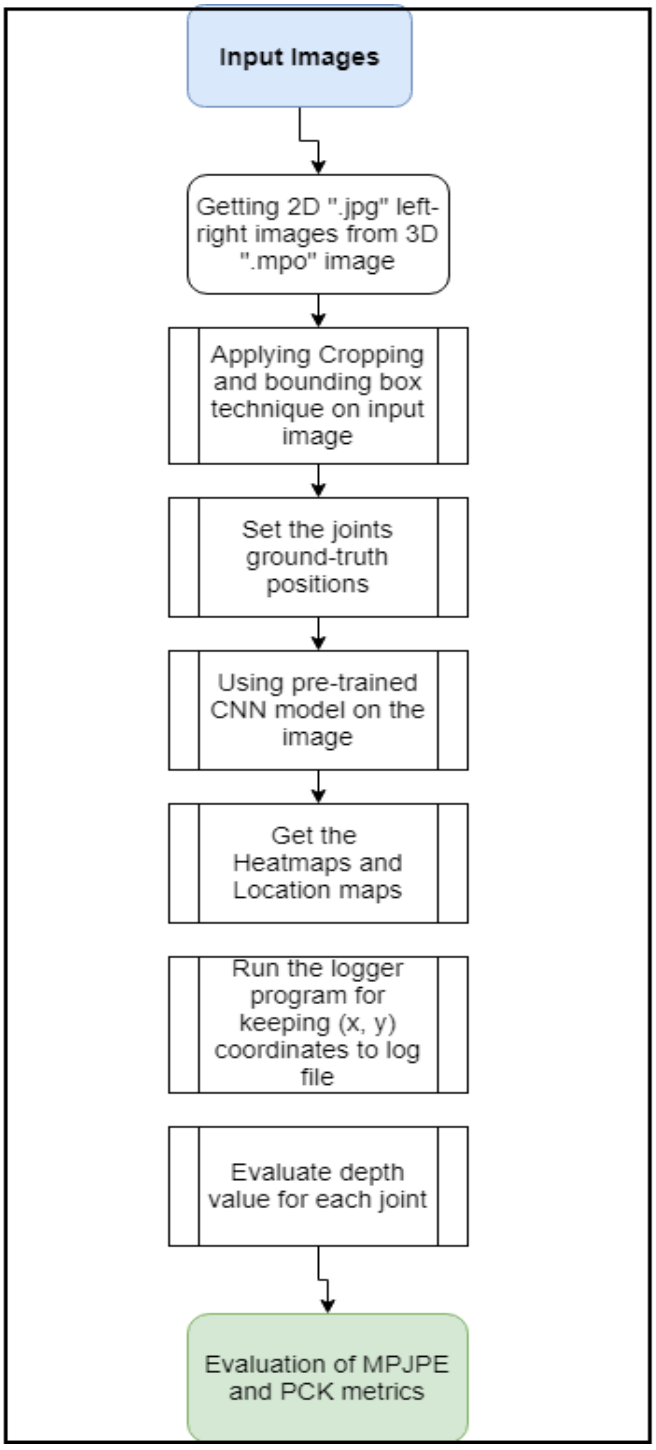
Figure 4.9: Images data created with different positions.



4.3 TESTING DATASET ON THE TRAINED MODEL

In this section, we explain our proposed method and discuss our implementations in addressing 3D pose estimation from 2D skeleton joint positions. Figure 4.10 illustrates the flowchart of our proposed method and implementations.

Figure 4.10: The Flowchart of our Proposed Method



All code development was done using the Matlab program. Our novel data set has 8 different positions and it is taken in outdoor environments. Image size is 3648 width and 2736 height pixels. Horizontal and vertical resolution is 72 dpi (Dots Per Inch)

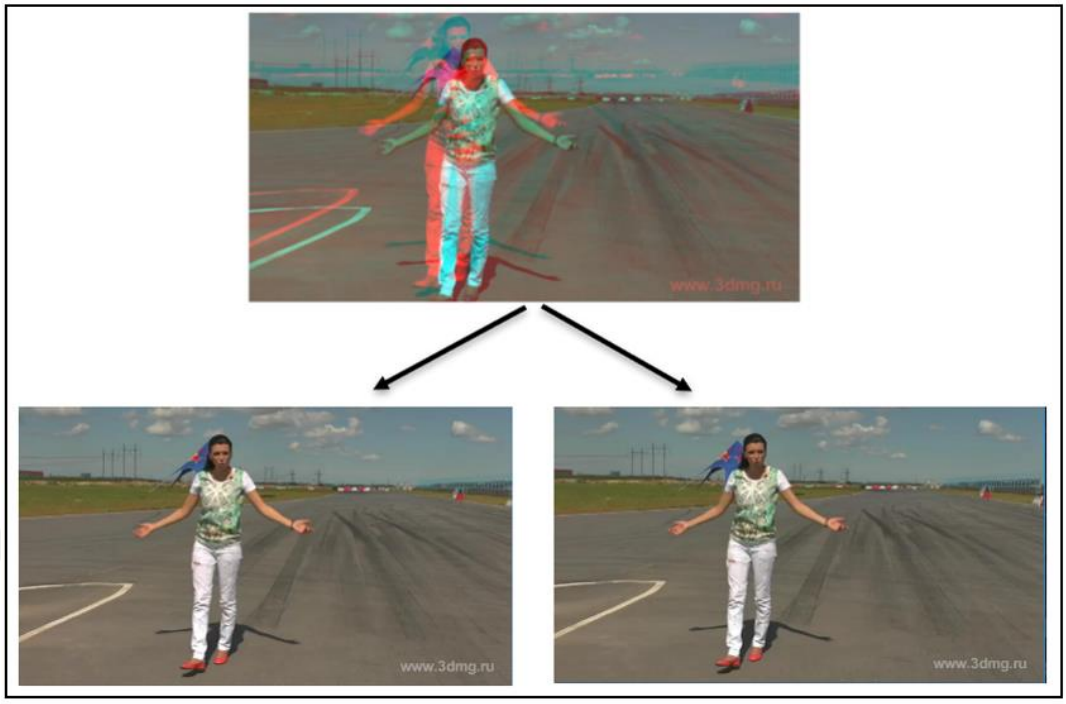
There are three main steps to train our data set to get 3D joint positions in (X, Y, Z) coordinate system:

a) Getting 2D left and right images from each 3D image,
b) Estimating skeleton joints positions with pre-trained CNN model,
c) Finding depth value from obtained coordinates.

In this section, all three parts are explained below.

Firstly, our data set's images have ".mpo" (Multi Picture Object File) extension. To successfully work with the model, we must separate our 3D stereo image into left and right 2D ".jpg format" images to run on the pre-trained model separately, as shown in Figure 4.11 (In this figure, the image is not used to evaluate our method because the image is not included in our set. It is just used for exemplary purposes). For that separation operation, the graphics viewer program (IrfanView).

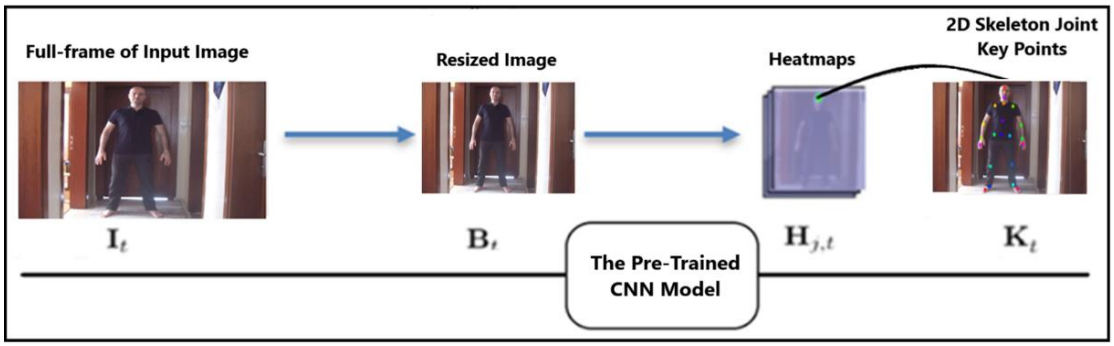
Figure 4.11: Separation 3D stereo images into 2D left and right image.



Source: Left and Right Images are used from SHPED - The Stereo Human Pose Estimation Data set

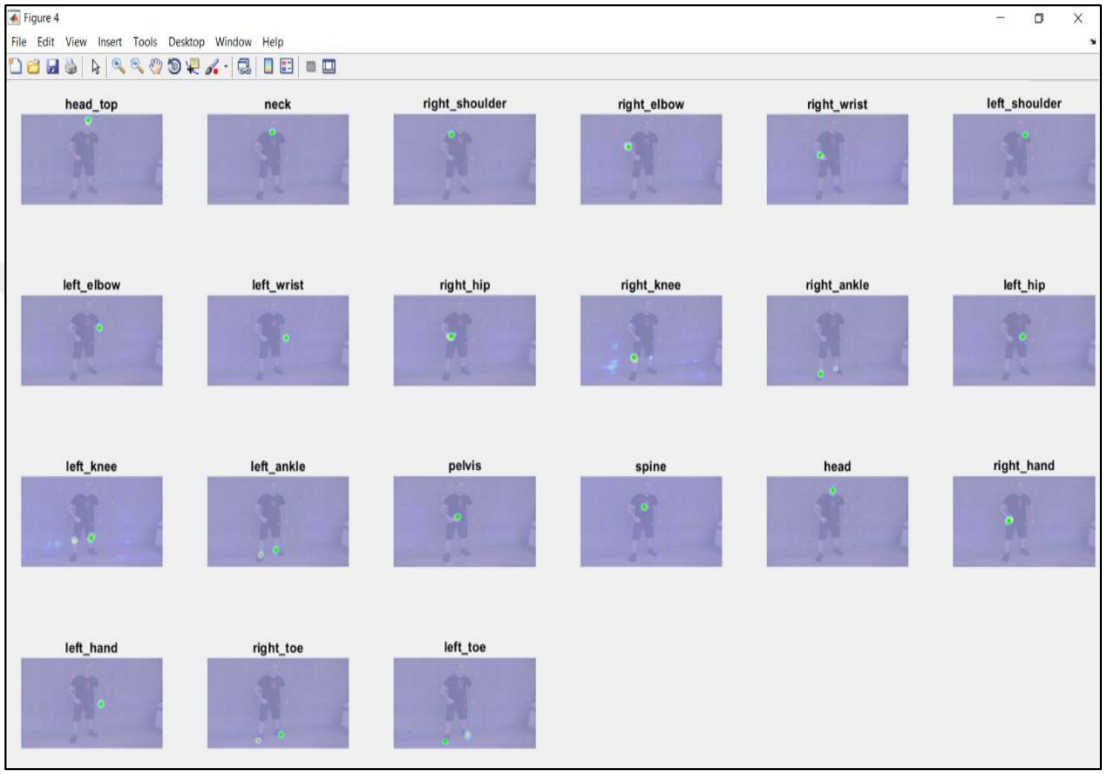
Secondly, after getting left and right images of our data set, we find and mark 2D skeleton joint points on the pictures with the pre-trained method in the referenced work. In Figure 4.12, it is showed that given an image I_t at frame t, the crop according to the person coordinates B_t is extracted by cropping the input image, using the previous frame's key points K_{t-1}. After cropping the image, the state-of-the-art CNN model jointly predicts 2D heatmaps $H_{j,t}$ for all 21 joints. The 2D key points K_t are retrieved from $H_{j,t}$ and, after filtering step.

Figure 4.12: Finding 2D points architecture from a given image.



The estimated 2D skeleton joint positions by heatmaps is shown in below for the "Discussion" image from our study:

Figure 4.13: Estimating 2D points by heatmap.



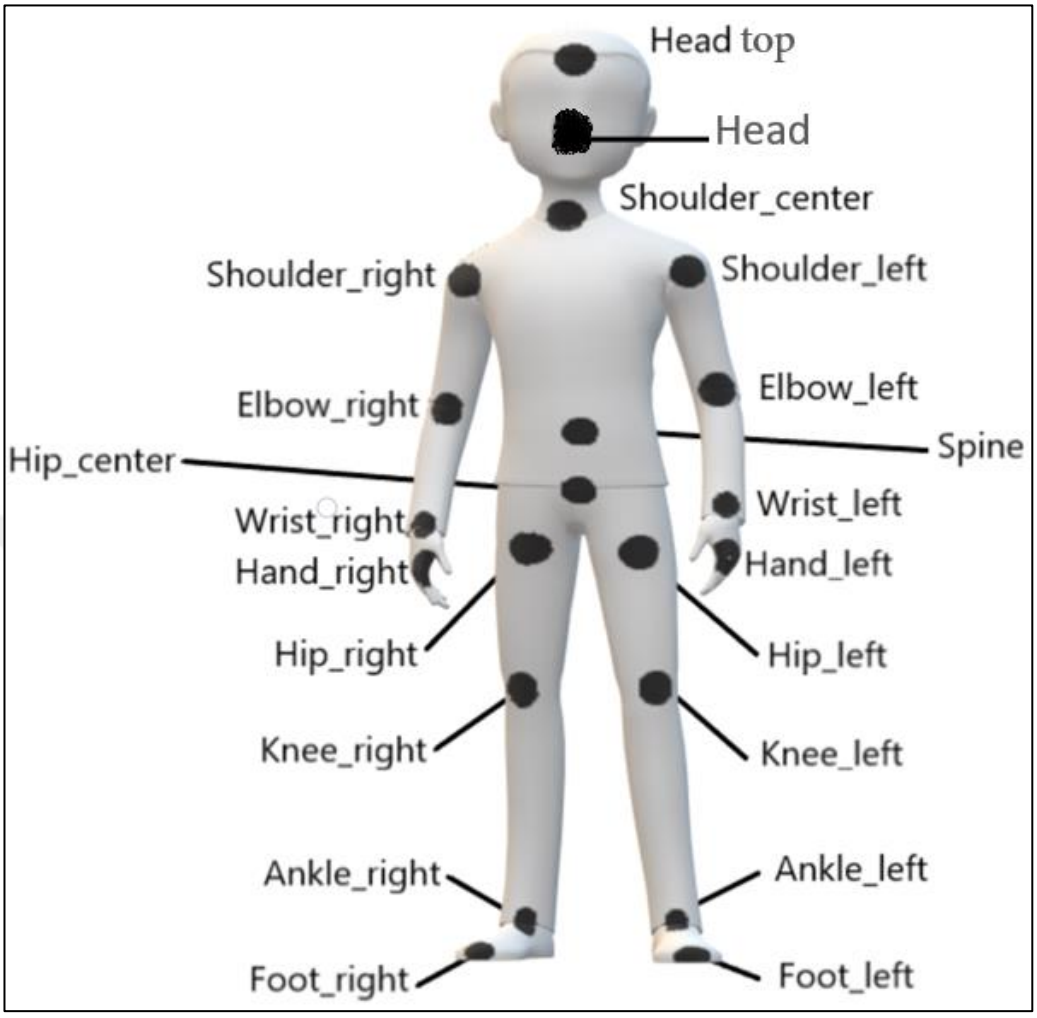
This is all heat maps for every joint for the first image of our dataset. When estimating some joint positions might be miscalculated for some action poses from our data set such as ankle, knee and toe positions in the figure 4.13.

In our study, the left and right image run separately. We developed a log file creator program that writes (X, Y) coordinates in a loop while the CNN model estimates 2D positions from heatmaps. The pre-trained CNN model architecture estimates 21 skeleton joints which is showed in Table 4.1 and Figure 4.14:

Table 4.1: Human Skeleton Joint Positions

1	Head Top
2	Neck
3	Right Shoulder
4	Right Elbow
5	Right Wrist
6	Left Shoulder
7	Left Elbow
8	Left Wrist
9	Right Hip
10	Right Knee
11	Right Ankle
12	Left Hip
13	Left Knee
14	Left Ankle
15	Pelvis
16	Spine
17	Head
18	Right Hand
19	Left Hand
20	Right Toe
21	Left Toe

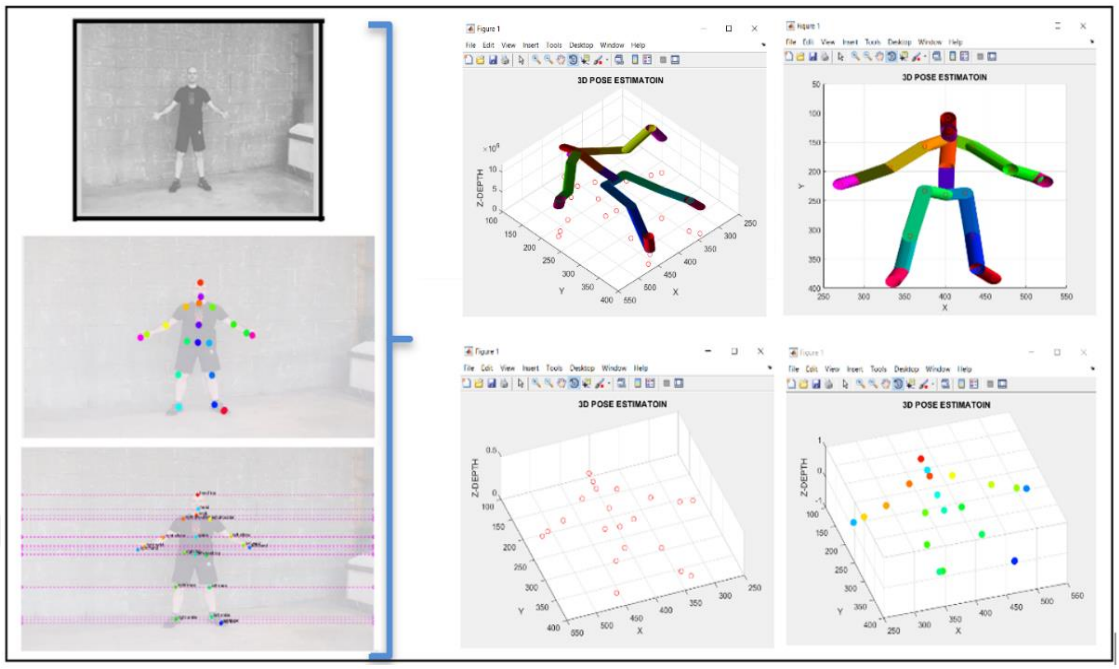
Figure 4.14: An Illustrator of Joint Positions on Human Body.



Thirdly, as explained in section 4.1.2, depth can be calculated for each skeleton joint with the equation of (10). Depth can be calculated using disparity. After the calculations, we get the depth value of each joint. Regarding disparity-based methods, (López-Quintero, 2016) propose an illustrated structure model able to estimate 2D poses on stereo image pairs extracted from the 3D image or an image frame sequence of a video.

After all, resizing the image from our dataset, then estimating 2D skeleton joint positions with the pre-trained CNN model and finding 3D joint positions with calculating depth information is shown in Figure 4.15 with Matlab Figure window.

Figure 4.15: Lifting 3D from 2D (x, y) coordinates and depth value.



4.4 EVALUATION AND ERROR MEASURE METRICS

Evaluation metrics explain the performance of a model. An important aspect of evaluation metrics is their capability to discriminate among model results. There are different types of measurement methods. Each method has its advantages and disadvantages. Because we compare our results with the referenced study (Mehta. 2017), as much of literature called as Percentage of Correct Parts (PCP) and Mean Per Joint Position Error (MPJPE) are used to compare and evaluate the success rates of the results (Ionescu, 2014), (Sapp, 2013).

To get results for MPJPE, there are two sub-process. The first one is per joint position error calculates separately with Euclidean distance between ground truth and prediction for a joint which has (X, Y, Z) coordinate. The second one, means of per joint position error for all k joints (in our study k is 21 joints). General formulation of MPJPE is shown in the equation of (12) below:

$$E_{mpjpe}(f, S) = \frac{1}{Ns} \sum_{i=1}^{Ns} ||m_{f,S}^{(f)}(i) - m_{gt,S}^{(f)}(i)||_2 \qquad (12)$$

Let $m_{f,S}^{(f)}(i)$ be a function that returns the coordinates of the i^{th} joint of skeleton S at frame f from the pose estimator f. And, let $m_{gt,S}^{(f)}(i)$ be the i^{th} joint of the ground truth frame f. N_s is the number of joints in skeleton S. The lower MPJPE result means lower error.

To calculate PCK, as first step the torso height is found. In this study, we take the distance between "right-hip" and "left-shoulder" as the torso height. In PCK method, the detected joint is considered correct if the distance between the predicted and the true joint is within a certain threshold value. This threshold can vary. According predicted and true joint positions, (13) shows the PCK calculation formula.

$$acc(r) = \frac{1}{N}\sum_{i=1}^{N}\left(\left|\left|y_i^p - y_i^g\right|\right|_2 < t * torso\right) \quad (13)$$

where N be the number of key points y_i^p is an array that holding points locations for the predicted each joint position. Likewise, y_i^g states the ground-truth joint positions. t is our threshold value. The higher PCK result means joints estimation percentage is higher.

5. EXPERIMENTS AND RESULTS

In this chapter, we are going to demonstrate experiments and results of our data set which has been used to run and evaluate in our study. Then, it will further discuss results between the state-of-the-art results and ours. We will compare them according to the ground-truth annotation each image set with respect to PCK and MPJPE metrics.

Our experiments ran on a computer with the following specifications,

i. Windows Pro 10 64-bit operating system,
ii. NVIDIA GeForce GTX 960M graphics card,
iii. Intel I7-6700 HQ CPU,
iv. 16 GBs of Ram

The tools (software and hardware) used for the experiment are,

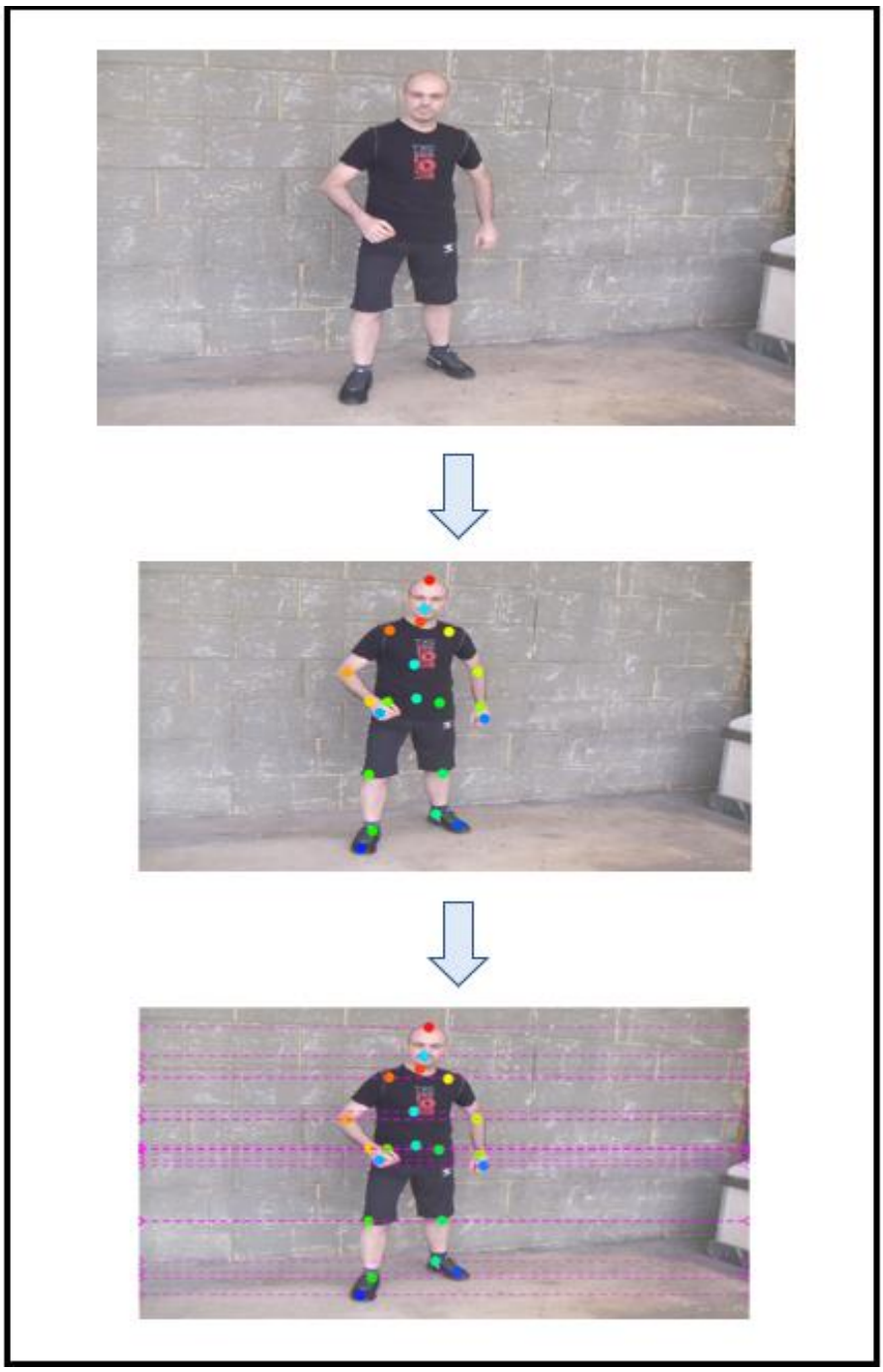
i. Matlab R2018a version
ii. Windows Caffe
iii. Python 3.5 version
iv. CUDA 7.5
v. Visual Studio 2015
vi. IrfanView
vii. Fujifilm FinePix Real 3D W1 model

As detailed in the section 4.2, a stereo image data set which contains images in different poses (walking, sitting, waiting, doing sport, taking a photo, posing, discussion and walking together) has been created. We process this data set in three main steps and compare the results of ground-truth, with our method's and referenced method's.

As first step, 3D images are separated left and right 2D images. For each position, all the pair of images are processes separately on the pre-trained model as described in the section 4.3. The pre-trained model uses the image cropping technique, then

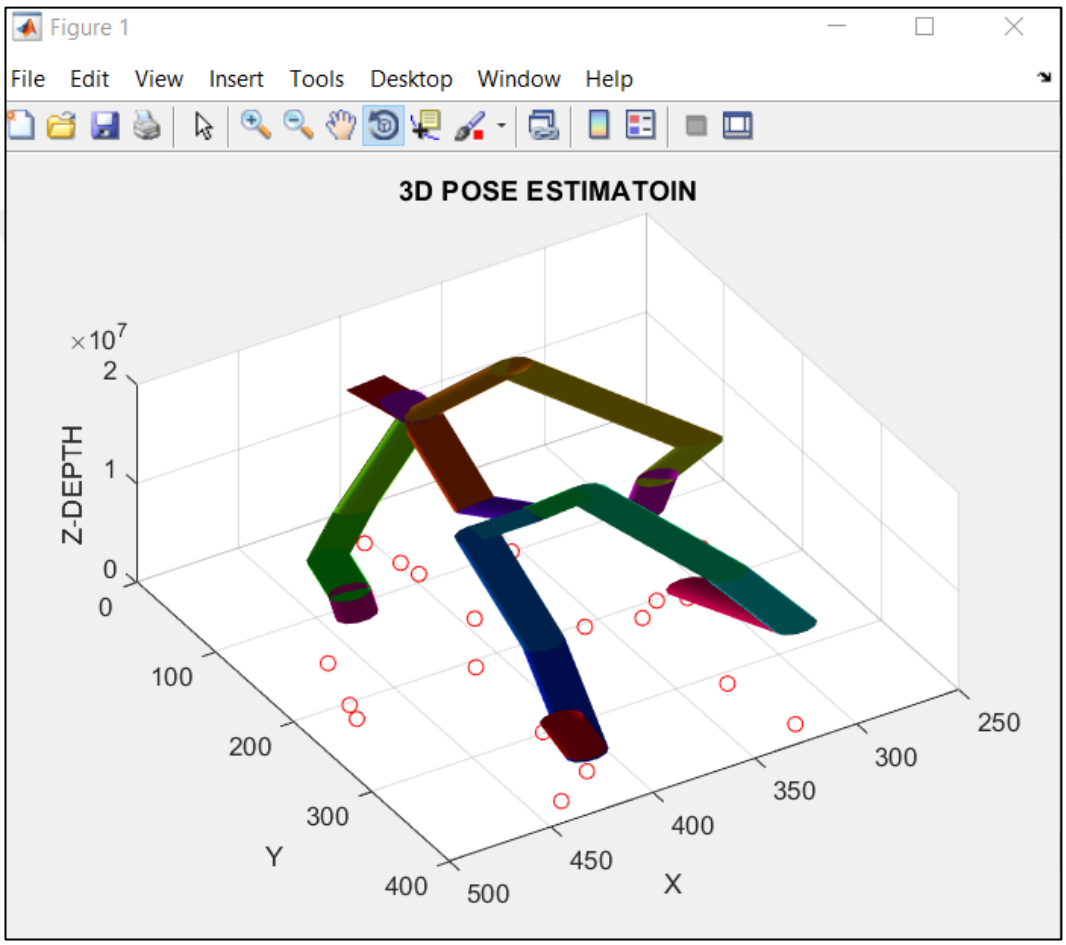
by CNN regression architecture estimates a heatmap which shows 21 joints positions. From these estimated joints, all 2D joints positions are estimated on the 2D images and we get x and y coordinates by our logger program at the same time.

Figure 5.1: Estimating 2D joint locations of stereo left and right image



As second step, as mentioned in the section 4.1.2, depth values for all 21 joints from left and right images are calculated, the difference of location value in pixels of the same joints between left and right images are called disparity. So, we obtain our (x, y) coordinates and depth results.

Figure 5.2: 3D estimating human poses – Matlab Figure result



As third step, we used the referenced study's pre-trained model for getting 2D results on our data set images. But we use the whole model to estimate 3D results. This way, we are able to compare the results of these two studies.

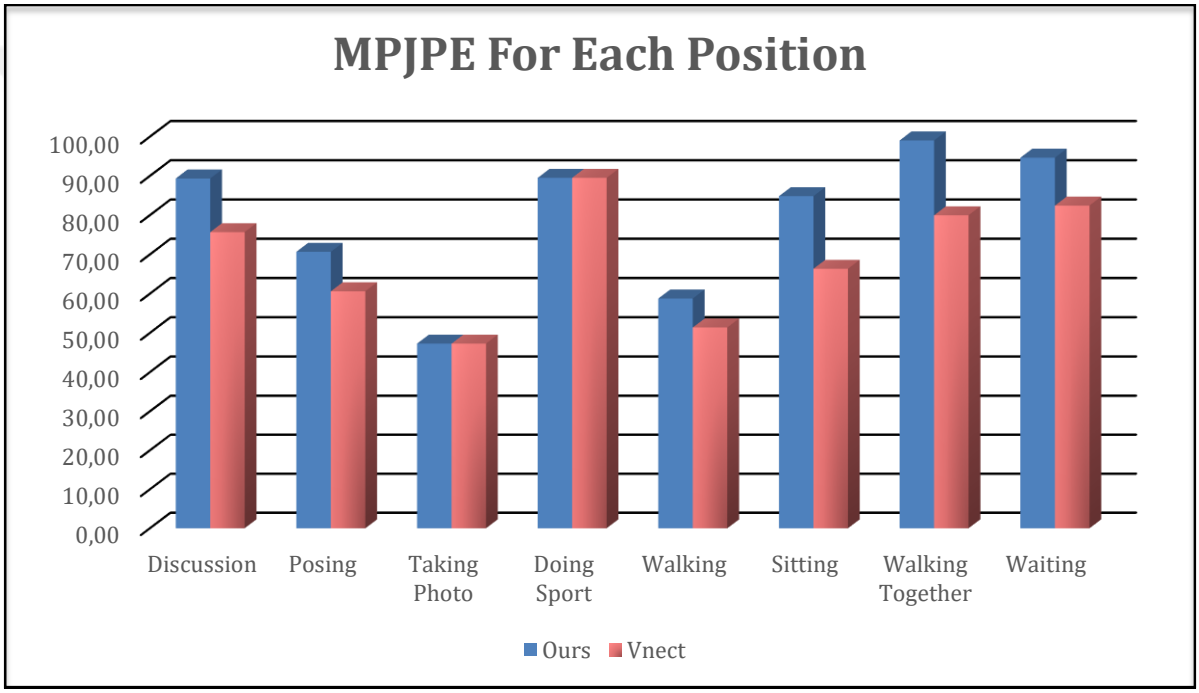
Using all these pixel values, we developed the code according to the MPJPE and PCK algorithms which are detailed in 4.4 section to measure the performance of

our model and the other one. The other one is called “Vnect” (Mehta, 2017). So, we use that name within result tables and figures.

Table 5.1: MPJPE results for each pose. The numbers are the mean per joint errors (mm) in 3D evaluated for different actions

Poses	Discussion	Posing	Taking Photo	Doing Sport	Walking	Sitting	Walking Together	Waiting
Ours	82.45	75.21	147.31	120.63	113.89	37.60	30.96	38.13
Vnect	85.60	81.95	**146.67**	134.88	122.74	55.96	42.21	49.05

Figure 5.3: MPJPE results for each action.



Lower result is better.

For eight different poses, we get the MPJPE results which is showed in Table 5.1 and Figure 5.3. According to these results, our model has higher accuracy than Vnect in most of the cases. “Walking Together” is the pose with the lowest MPJPE, which means the difference between the predicted and ground truth skeleton joint locations are less than the results for the other poses.

The “Taking Photo” pose has the lowest accuracy (highest MPJPE) among all the poses. We also can see that for this pose, our method has lower accuracy than to Vnect.

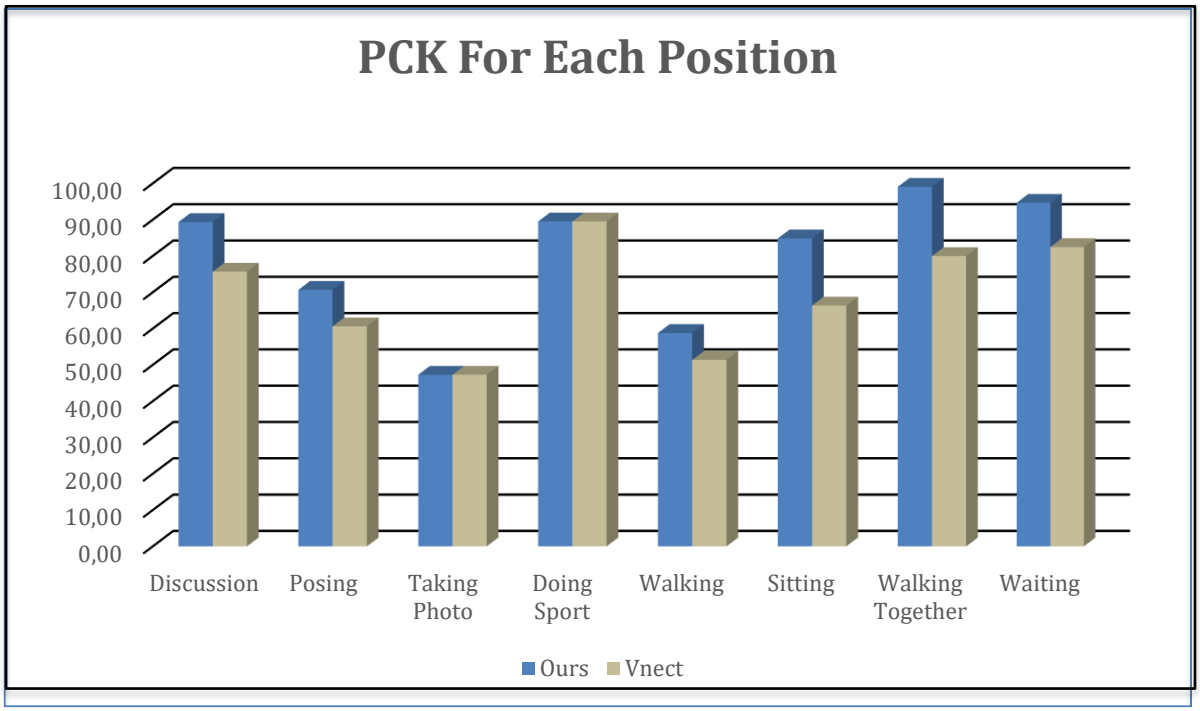
The highest difference by ratio between 2 methods was observed for the image in “Sitting” pose. Our method’s MPJPE value is almost %45 lower than Vnect’s result.

The lowest difference by ratio between 2 methods (where our method has a lower MPJPE value) was observed for the image in “Posing” pose. These values show that the success of these methods is heavily dependent on the pose.

Table 5.2: PCK results for each pose. The numbers are the percentage of correct key-points in 3D evaluated for different actions.

Poses	Discussion	Posing	Taking Photo	Doing Sport	Walking	Sitting	Walking Together	Waiting
Ours	89.34	70.66	47.27	89.51	58.77	84.85	99.05	94.65
Vnect	75.68	60.61	47.27	89.51	51.39	66.34	79.98	82.44

Figure 5.4: PCK results for each action.



Higher result is better.

For eight different poses, we get the PCK results which is showed in Table 5.2 and Figure 5.4. Threshold is chosen 0.2 in our study like in most of other studies. PCK@0.2 is the distance between predicted and true joint which is lower than 0.2 * torso diameter. According to these results, our model has higher accuracy than Vnect in most of the cases. “Walking Together” is the pose with the highest PCK, which means the difference between the predicted and ground truth skeleton joint locations are less than the results for the other poses.
The “Taking Photo” pose has the lowest accuracy among all the poses. We also can see that for this pose and “Doing Sports” pose, our method has equal accuracy to Vnect.

The highest difference by ratio between 2 methods was observed for the image in “Sitting” pose. Our method’s PCK value is almost %19 higher than Vnect’s result.

These values show that the PCK metric results of these methods is heavily dependent on the pose.

All in all, we see that PCK and PMJPE performance measurement results are proportional. According to these results, we achieve our aim to estimating 3D human pose successfully than the referenced study in most cases.

6. DISCUSSION AND CONCLUSION

In addition to discussing the possibilities of future studies, this chapter reviews the results of this research.

In this research, we have introduced a new method on top of the referenced study, called “Vnect” (Mehta, 2017) that obtains the 3D human kinematic skeleton pose estimation from 2D pose with a new image data set. For this purpose, we created our novel limited stereo data set consists of eight different action poses with three different images for each pose (24 images) using Fujifilm FinePix REAL 3D W1 camera. Then, we have used this data set with the referenced study and added a well-known, old but still widely used computer vision method, called a stereo matching method. We use this method for getting a 3D human pose from 2 (left and right) 2D images which we use to find joint positions using the shared CNN model and weights (http://gvv.mpi-inf.mpg.de/projects/VNect/).

After getting the 2D joint positions using our model, we calculate the depth value for each different image with stereo matching technique. Then we compare the results of our approach with the result of the referenced study. To compare the accuracy, we use two different performance metrics, MPJPE and PCK. The values of these metrics correlate with each other and our method has higher accuracy for most of the images used in this study.

In conclusion, we found that our method was able to give better results than the referenced study when estimating the depth using 3D stereo camera scenes.

With the increase in the number of studies and technological developments in this field, better results and performances can be achieved. So, it can be used widely and much more accurate in real-life applications such as medical purposes, security purposes, education, sports, entertainment, etc.

To improve this study further, the CNN model can be trained using a data set with more annotated data. This way, it would have more accurate results. The implemented model of our study supports images that contain a single person. The model can be improved to be able to process more than one human body skeleton pose estimation if a multi-person data set can be gathered and used for training the model.

Due to hardware limitations this study cannot process videos in real time. A video consists of image sequences. With a more advanced system, this study can be used to estimate human pose in videos.

We created our own data set because of the lack of 3D stereo image dataset. There are more stereo image data sets available, but they have no information of the camera used to create them. In order to calculate the depth information, camera baseline and focal length information must be known. In the future there might be more data sets available with the needed attributes given. With more data contains various background distributions, light directions, poses and shapes the results can be more meaningful and accurate for different poses.

REFERENCES

Books

Bradski, G., Adrian, K., 2008. *A. Learning OpenCV: Computer Vision with the OpenCV Library*, O'Reilly Media, Inc.: Sevvan, CA, USA.

Fua, P., 1993. *A parallel stereo algorithm that produces dense depth maps and preserves image features*. Machine vision and applications, 6(1) :35–49.

Grosky, W. & Tamburino, LA. 1990. *A unified approach to the linear camera calibration problem*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 12(7):663 –671.

Hartley, R., & Zisserman, A., 2003. *Multiple View Geometry in Computer Vision.* New York: Cambridge University Press.

Krizhevsky, A., Sutskever, I., Hinton, G., 2017. *ImageNet classification with deep convolutional neural networks*. Communications of the ACM. 60 (6): 84–90. doi:10.1145/3065386. ISSN 0001-0782.

Laganière R., 2011. *OpenCV 2 Computer Vision Application Programming Cookbook:Over 50 Recipes to Master This Library of Programming Functions for Real-Time Computer Vision*, Packt Publishing Ltd.

Minsky, M., & Papert, S., 1969. *Perceptrons*, MIT Press, Cambridge. pp.780-782.

Moeslund, T.B., Hilton, A., Krüger, V., Sigal, L., 2011. *Visual Analysis of Humans*, Springer.

Mühlmann, K., Dennis, M., Jürgen, H., & Reinhard, M., 2002. *Calculating Dense Disparity Maps from Color Stereo Images, an Efficient Implementation*, International Journal of Computer Vision, Vol. 47, No. 1, pp.79-88.

Schmidhuber, J., 2015. *Deep learning in neural networks: an overview*. Neural Networks 61, Elsevier Ltd., pp. 85-117.

Shapiro, L. G. & Stockman, G. C., 2001. *Computer vision*. Prentice Hall.

Sonka, M., Hlavac, V. & Boyle, R., 2007. *Image Processing, Analysis, and Machine Vision*. London: Thomson-Engineering.

Werbos, P., 1982. *Applications of advances in nonlinear sensitivity analysis*. Springer. pp.762–770.

Periodicals

Andriluka, M., Roth, S., Schiele, B., 2012. *Discriminative appearance models for pictorial structures*. International Journal of Computer Vision 99(3).

Atick J.J., Griffin P. A., and Redlich A. N., 1996. *Statistical Approach to Shape from Shading: Reconstruction of 3D Face Surfaces from Single 2D Images*, Computation in Neurological Systems, vol. 7, no. 1.

Baak, A., Müller, M., Bharaj, G., Seidel, H.-P., & Theobalt, C., 2011. *A datadriven approach for real-time full body pose reconstruction from a depth camera*. In IEEE International Conference on Computer Vision.

Besl P. J., 1988. *Geometric Modeling and Computer Vision*, Proc. IEEE, vol. 76, no. 8, pp. 936-958.

Blake, A., & Zisserman, A., 1987. *Visual Reconstruction*, MIT Press.

Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., & Black, M. J., 2016. *Keep it SMPL: Automatic estimation of 3d human pose and shape from a single image*. In European Conference on Computer Vision, Springer, pp. 561–578

Bulat, A., & Tzimiropoulos, G., 2016. *Human pose estimation via convolutional part heatmap regression*. In European Conference on Computer Vision.

Cao, Z., Simon, T., Wei, S., & Sheikh, Y., 2016. *Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields*. arXiv preprint arXiv:1611.08050.

Farabet C., Couprie C., Najman L. and LeCun Y., 2013. *Learning hierarchical features for scene labeling*, Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 35, no. 8, pp. 1915–1929.

Ferrari, V., Marin-Jimenez, M., & Zisserman, A., 2009. *Pose search: retrieving people using their pose*. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8

Gonzalez, R., Woods, & Richard E. 2008. *Chapter 2: Moving Object Detection & Tracking in Videos*. Digital image processing. 3rd edn. Upper Saddle River, N.J.: Prentice Hall.pp.15-39.

Hashim, Y., Umar, I., Bjorn, K., Andreas, W., & Juergen, G. June 2016. *A dual-source approach for 3d pose estimation from a single image*. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Hinton G. E., Srivastava N., Krizhevsky A., Sutskever I and Salakhutdinov R. R., 2012. *Improving neural networks by preventing co-adaptation of feature detectors*.

Hinton, G. E., Dayan, P., Frey, B. J., and Neal, R., 1995. *The wake-sleep algorithm for self-organizing neural networks*. Science, 268, pp. 1158–1161.

Hinton, G. E., Osindero, S., Teh, Y.W., 2006. *A fast learning algorithm for deep belief*

nets. Neural Computation. 18 (7), MIT Press Cambridge. pp.1527-1554.

Horn, B. K. P. 1986. *Robot Vision MIT Electrical Engineering and Computer Science*. The MIT Press, mit press ed edition.

Howard, IP., & Rogers, BJ., 2002. *Seeing in Depth Volume 2 Depth Perception*, Oxford University Press

Iqbal, U., Milan, A., & Gall, J., 2017. *PoseTrack: Joint Multi-Person Pose Estimation and Tracking*, Computer Vision Foundation (CVPR).

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L., 2014. *Large-scale Video Classification with Convolutional Neural Networks*. IEEE Conference on Computer Vision and Pattern Recognition, 10.1109/CVPR.2014.223, USA.

López-Quintero, M.I., Mar´ın-Jim´enez, M.J., Mu˜noz-Salinas, R., Madrid-Cuevas, F.J., Carnicer, R.M., 2016. *Stereo pictorial structure for 2d articulated human pose estimation*. Machine Vision and Applications 27, pp. 157–174.

LeCun, Y., Bengio, Y., Hinton, G., 2015. *Deep learning*. Nature 521, pp. 436–444. http:// dx.doi.org/10.1038/nature14539.

Lo´pez-Quintero, M.I., Marı´n-Jimenez, M.J., Munoz-Salinas, R., Medina-Carnicer, R., 2017. *Mixing body-parts model for 2d human pose estimation in stereo videos.* IET Computer Vision 11.

Mahendran, S., Ali, H., Vidal, R., 2018. *Convolutional Networks for Object Category and 3D Pose Estimation from 2D Images*. In book: Computer Vision – ECCV 2018 Workshops.

May S., Pervoelz K., Surmann H., 2007. *Chapter 11: 3D Cameras: 3D Computer Vision of Wide Scope*, Vision Systems - Applications, I-Tech Educ. and Publ., pp. 181-202.

Mehta, D., Srinath S., Oleksandr S., Helge R., Mohammad S., Hans-Peter S., Weipeng X., Dan C., and Christian T., 2017. *VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera*, vol. 36.

Mehta, D., Rhodin, H., Casas, D., Sotnychenko, O., Xu, W., & Theobalt. C., 2016. *Monocular 3D Human Pose Estimation in The Wild Using Improved CNN Supervision*. arXiv preprint arXiv:1611.09813v2.

Nie, B. X., Wei, P., & Zhu, S., 2017. *Monocular 3D human pose estimation by predicting depth on joints*. International Conference on Computer Vision.

Romero, A., Gatta, C., Camps-valls, G., Member, S., 2016. *Unsupervised deep feature extraction for remote sensing image classification*. IEEE Trans. Geosci. Remote Sens. 54, pp. 1349–1362.

Rosenblatt, F., 1958. *The perceptron: A probabilistic model for information storage and organization in the brain*. Psychological Review. 65 (6), pp.386-408.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. *Learning representations by backpropagating errors*. Nature. 323, pp. 533–536.

Shingade, A., Ghotkar, A., 2014. *Animation of 3D Human Model Using Markerless Motion Capture Applied to Sports*, Pune Institute of Computer Technology.

Sichkar V., 2018. *Convolutional Neural Networks for Image Classification with CIFAR-10 dataset*.

Tekin, B., Katircioglu, I., Salzmann, M., Lepetit, V., & Fua, P., 2016. *Structured Prediction of 3D Human Pose with Deep Neural Networks*. In British Machine Vision Conference (BMVC).

Tekin, B., Rozantsev, A., Lepetit, V., & Fua, P., 2016. *Direct Prediction of 3D Body Poses from Motion Compensated Sequences*. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Tekin, B., Márquez-Neila, P., Salzmann, M., & Fua, P., 2016. *Fusing 2D Uncertainty and 3D Cues for Monocular Body Pose Estimation*. arXiv:1611.05708.

Tinchcombe, M., 1989. *Multilayer Feedforward Networks are Universal Approximators*., Neural Networks, Vol. 2, USA, pp. 359–366.

Tompson, J., Goroshin, R., Jain, A., LeCun, Y., & Bregler, C, 2015. *Efficient object localization using convolutional networks*. IEEE Conference on Computer Vision and Pattern Recognition.

Tong, J., Zhou, J., Liu, L., Pan, Z., & Yan, H., 2012. *Scanning 3d full human bodies using kinects*. Visualization and Computer Graphics, IEEE Transactions on 18.4. pp. 643-650.

Toshev, A., & Szegedy, C., 2014. *Deep pose: Human pose estimation via deep neural networks*. CoRR, IEEE Conference on Computer Vision and Pattern Recognition.

Xu J., Yi Q., Fu C., Yin H., Zhao Z., Chen K. 2011. *Chapter Active Stereo Vision for 3D Profile Measurement, Advances in Stereo Vision,* InTech, pp. 1-16.

Yasin, H., Iqbal, U., Krüger, B., Weber, A., & Gall, J., 2016. *A Dual-Source Approach for 3D Pose Estimation from a Single Image.* In Conference on Computer Vision and Pattern Recognition (CVPR).

Zhang W., Liu Z., Zhou L., Leung H., Chan A., 2017. *Martial Arts, Dancing and Sports dataset: A challenging stereo and multi-view dataset for 3D human pose estimation*., Image and Vision Computing, Volume 61 Issue C, pp. 22-39.

Zhang, Z., 2000. *A flexible new technique for camera calibration*. IEEE Trans. Pattern Anal. Mach. Intell. 22, No. 11, pp. 1330–1334.

Zhou, X., Zhu, M., Leonardos, S., Derpanis, K., & Daniilidis, K., 2015. *Sparseness Meets Deepness: 3D Human Pose Estimation from Monocular Video*. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C., 2014. *Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments*. IEEE Transactions On Pattern Analysis And Machine Intelligence, pp. 4-8.

Sapp, B., Taskar, B., 2013. *MODEC: Multimodal Decomposable Models for Human Pose Estimation*. IEEE Conference on Computer Vision and Pattern Recognition, pp. 5-7.

Other Publications

MathWorks, *Convolutional Neural Network,* https://www.mathworks.com/solutions/deep-learning/convolutional-neural-network.html [accessed 4 February 2019]

Microsoft, Developer Page, 2014, Kinect for Windows SDK 2.0 https://developer.microsoft.com/en-us/windows/kinect [accessed 4 February 2019]

Karpathy, A., Convolutional Neural Networks for Visual Recognition (Course Notes). url: http://cs231n.github.io/. [accessed 21 December 2018]

Stanford Courses, *Neural Networks History,* https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/History/history1.html [accessed 7 January 2019]

Zeng, W., 2012. Microsoft Kinect Sensor and Its Effect. https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/Microsoft20Kinect20Sensor20and20Its20Effect20-20IEEE20MM202012.pdf [accessed 3 January 2019]

Youtube, yt3d:enable=true keyword for stereo videos https://www.youtube.com [accessed 24 Novermber 2018]

IrfanView, Graphic Viewer Program *https://www.irfanview.com* [accessed 24 Novermber 2018]