

Spring 2023

GENDER CLASSIFICATION VIA HUMAN JOINTS USING CONVOLUTIONAL NEURAL NETWORK

Cheng-En Sung
San Jose State University

Follow this and additional works at: https://scholarworks.sjsu.edu/etd_projects



Part of the [Artificial Intelligence and Robotics Commons](#), and the [Other Computer Sciences Commons](#)

Recommended Citation

Sung, Cheng-En, "GENDER CLASSIFICATION VIA HUMAN JOINTS USING CONVOLUTIONAL NEURAL NETWORK" (2023). *Master's Projects*. 1208.

DOI: <https://doi.org/10.31979/etd.hrw8-9n8k>

https://scholarworks.sjsu.edu/etd_projects/1208

This Master's Project is brought to you for free and open access by the Master's Theses and Graduate Research at SJSU ScholarWorks. It has been accepted for inclusion in Master's Projects by an authorized administrator of SJSU ScholarWorks. For more information, please contact scholarworks@sjsu.edu.

GENDER CLASSIFICATION VIA HUMAN JOINTS USING CONVOLUTIONAL
NEURAL NETWORK

A Project

Presented to

The Faculty of the Department of Computer Science

San José State University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

By

Cheng-En Sung

May 2023

© 2023

Cheng-En Sung

ALL RIGHTS RESERVED

The Designated Thesis (or Dissertation) Committee Approves the Thesis Titled

GENDER CLASSIFICATION VIA HUMAN JOINTS USING CONVOLUTIONAL
NEURAL NETWORK

By

Cheng-En Sung

APPROVED FOR THE DEPARTMENT OF COMPUTER SCIENCE

SAN JOSÉ STATE UNIVERSITY

April 2023

Nada Attar, Ph.D. Department of Computer Science

Reem Albaghli, Ph.D. Department of Computer Science

Philip Heller, Ph.D. Department of Computer Science

ABSTRACT

With the growing demand for gender-related data on diverse applications, including security systems for ascertaining an individual's identity for border crossing, as well as marketing purposes of digging the potential customer and tailoring special discounts for them, gender classification has become an essential task within the field of computer vision and deep learning. There has been extensive research conducted on classifying human gender using facial expression, exterior appearance (e.g., hair, clothes), or gait movement. However, within the scope of our research, none have specifically focused gender classification on two-dimensional body joints. Knowing this, we believe that a new prediction pipeline is required to improve the accuracy of gender classification on purely joint images. In this paper, we propose novel yet simple methods for gender recognition. We conducted our experiments on the BBC Pose and Short BBC pose datasets. We preprocess the raw images by filtering out the frame with missing human figures, removing background noise by cropping the images and labeling the joints via the C5 (model applied transfer learning on the ResNet-152) pre-trained model. We implemented both machine learning (SVM) and deep learning (Convolution Neural Network) methods to classify the images into binary genders. The result of the deep learning method outperformed the classic machine learning method with an accuracy of 66.5%.

***Keywords* - Gender Classification, Image pre-processing, Machine Learning, Deep Learning, Convolution Neural Network [CNN].**

ACKNOWLEDGMENTS

The success of this research is the result of the efforts of many individuals. First, I would like to express my special thanks to my lovely family, who always supported me along the way. Next, I would like to convey my appreciation to my project advisor, Professor Nada Attar, who always gives me valuable feedback and encourage me when my progress stock

TABLE OF CONTENTS

ABSTRACT	I
ACKNOWLEDGMENTS	II
TABLE OF CONTENTS	III
LIST OF FIGURES	IV
LIST OF TABLES	V
I. INTRODUCTION	1
II. RELATED WORKS	2
2.1 FACIAL EXPRESSION-BASED GENDER CLASSIFICATION	2
2.2 GAIT-BASED GENDER CLASSIFICATION	3
2.3 JOINT-BASED GENDER CLASSIFICATION	4
III. DATASET	6
3.1 DATASET DESCRIPTION	6
3.2 IMAGE PREPROCESSING	7
IV. METHODOLOGY	15
4.1 SVM	15
4.2 NN BACKGROUND	16
4.3 BUILDING CNN MODEL	23
V. EXPERIMENT	27
VI. CONCLUSION AND FUTURE WORK	35
VII. REFERENCE	37

LIST OF FIGURES

Figure 1: Sample images from BBC Pose and BBC Short Pose datasets	6
Figure 2: Sample image with missing human figure	7
Figure 3: Non-centralize human figure	8
Figure 4: Background Noise	9
Figure 5: Cropping images with customized ratio	10
Figure 6: C5 Network architecture	11
Figure 7: Hourglass Network architecture	12
Figure 8: CPN Network architecture	12
Figure 9: Image preprocessing with the image cropping and the joint prediction	14
Figure 10: SVM	15
Figure 11: 3x3 Filter (Kernel)	18
Figure 12: Stride	19
Figure 13: Padding	20
Figure 14: Pooling layer	21
Figure 15: Fully connected (Dense) layer	22
Figure 16: Dropout layer	23
Figure 17: Shallow CNN Structure	24
Figure 18: Deep CNN Structure	25

LIST OF TABLES

Table 1. BBC Pose and Short BBC Pose metadata	5
Table 2: Shallow CNN Architecture	24
Table 3: Deep CNN Architecture	25
Table 4: Snapshot of training and validation results on epochs 10, 20, and 30	27
Table 5: Loss and accuracy in figure format	28
Table 6: Testing Result (accuracy, recall precision, and F1 score) on four models	30
Table 7: Confusion Matrix	31

I. INTRODUCTION

Gender classification was first treated as a psychophysical research problem in the early twenty century [1,2] and has been widely investigated in recent decades. Undoubtedly, gender classification plays a significant role in many aspects nowadays. For example, it can be used to identify and address issues of discrimination and inequality based on gender confusion [3]. Besides, Gender classification could also be used for many commercial domains' applications such as human-computer interaction, computer vision, and computer-aided psychophysical analysis. In addition, gender classification is crucial for scientific research and data collection purposes. The study of gender classification is mainly focused on identifying the key features between male and female individuals and precisely distinguishing one from the other. While previous research mostly concentrated on gender classification via the hints from facial features [4,5] and/or exterior appearance (e.g., hair, clothes), we aim to present a novel way to predict an individual's gender solely via his/her body joints using state-of-the-art computer vision techniques along with the deep learning models. We plan to make a comparison between the prediction result of gender classification using human posture with the classic machine learning method and the Convolutional Neural Network (CNN) model that trains on the original 2-D RGB images.

II. RELATED WORKS

In this section, we are going to review prior study pertaining to various methodologies used for gender identification.

2.1 Facial Expression-based Gender Classification

M. S. Fathollahi et. al., [6] proposed an automatic gender classification system via two parallel CNN models using facial images. One of the two networks named CDCN uses the central difference convolution layer, while the other network, VCNN, employs vanilla CNN layers. The result of the two networks is then concatenated into a dense layer for classification. The whole system was trained on the Casia WebFace dataset. For testing, the model was evaluated on two datasets with – LFW (labeled faces in the wild) and FEI datasets. The method suggested by the authors obtain a rate of 97.79% accuracy for the LFW dataset and 99.10% for the FEI dataset.

A. Lahariya et. al., [7] proposed an ensemble CNN model by averaging the result of Mini-Xception [8] and a simple 4-layer vanilla CNN model for real-time emotion and gender detection. The proposed method successfully reach 68% accuracy for emotional classification into 7 categories (disgust, angry, fear, happy, sad, surprise, and neutral) on the FER-2013 dataset and 95% for binary gender classification on the IMDB dataset.

2.2 Gait-based Gender Classification

C. Xu et. al., [9] invented a unified framework for real-time gait-based age estimation. The proposed framework comes with the ability to classify gender using solely a single image. Compared to other existing methods, latency in video capturing could be largely reduced. To deal with the issue where a single image lacks motion information, the author suggested two state-of-the-art network structures - a gait cycle reconstruction network and a gait recognition network. The input image is first reconstructed to a gait cycle of a silhouette sequence through a gait-cycle network. The image sequence after processing is then passed through a gait recognition network for feature representation learning. The output of the gait recognition network is then used to predict the class of the gender and estimate the age of the input image. The author also claims that the proposed method is applicable to gait images with an arbitrary view, which is closer to real-world scenarios. The authors claim their work is the first research that address age estimation and gender classification using merely a single gait image, they compared their proposed method with two other models: GaitSet, a baseline model which predict the gender without reconstruct the gait cycle, and GEINet, the state-of-the-art age estimation method. The correctness of gender classification and age estimation was measured by the mean correct classification (CCR) and mean absolute error (MAE) respectively. In result, the proposed model attained 94.27% classification rate that surpassed the GaitSet model of 92.72%, and achieved 8.39 MAE, which was lower than the GEINet of 9.11.

2.3 Joint-based Gender Classification

2.3.1 3-D Joints

M. Azhar et al., [10] proposed a logistic-regression-based machine learning model utilizing whole body joints obtained by the Kinect sensor for gender classification. The proposed method involves several steps including 3D gait feature extraction via Kinect sensors, feature selection using statistical tools such as Cronbach's alpha, correlation, t-test, and ANOVA, and ultimately, classification is performed hinging on the selected feature using a binary logistic regression model. The proposed model successfully achieves up to 98% accuracy in gender classification using all body joints.

2.3.2 2-D Joint

To the best of our exploration, there is no previous study that predicts the individual's gender merely using 2-D human joint images. Previous works either taking the advantage of hints from facial expressions, hair, clothes [6][7], or gait [9], while another study that applies 3-D joint information for gender classification collected by Kinect sensor [10], still utilizes more prior knowledge than our research. This further emphasizes the challenges and significance associated with undertaking research. We now review past research for annotating the 2-D joints' position of the human body part.

J. Charles et. al, [11] presented a random forest regression model to predict the joint positions (shoulders, elbows, wrists) that do not require any preliminary knowledge from manual annotation and was able to perform real-time posture tracking on people that have not been seen before. As a result, this model achieved an 83.4% accuracy upon the predicted label within 5 pixels away from ground truth on 1,000 testing frames. In the same research, J. Charles et. al [12] release the qualitative BBC human pose dataset (in the format of a series of consecutive images). Newell, A. et. al, [13] devised a novel network structure – Hourglass, which produces high-resolution feature maps that have dominated the MPII benchmark and achieved a 67.1 Average Precision (AP) score on the COCO val2017 dataset. B. Xiao [14] et. al, presented an innovative network structure to track and estimate human pose by simply appending three deconvolutional layers to the last convolution stage in the ResNet and outperformed Hourglass by achieving up to 72.0 AP score on COCO val2017 dataset.

III. DATASET

3.1 Dataset Description

The VGG human pose estimation dataset is a collection of video datasets that have been annotated with the human upper-body gesture. Among these datasets, the BBC Pose and BBC Short Pose were well-established datasets that were organized and released by J. Charles et. al [11][12] at the Visual Geometry Group (VGG) at the University of Oxford. It contains 25 broadcast videos (about 44 GB, 1.5 M frames), each lasting between 0.5 hours to 1.5 hours (4K frames on average) that feature recordings from the BBC with sign language interpretation added. Table 1 presents the basic topological features of the BBC Pose and Short BBC Pose datasets.

Table 1. BBC Pose and Short BBC Pose metadata

Dataset	BBC Pose	Short BBC Pose
Total videos	20	5
People in video dataset	9	5
Frames	1.5 M	380 K

This dataset is commonly used for training and evaluating computer vision models for tasks such as human pose tracking. Fig. 1 demonstrates the images sampled from the BBC Pose and Short BBC Pose datasets.

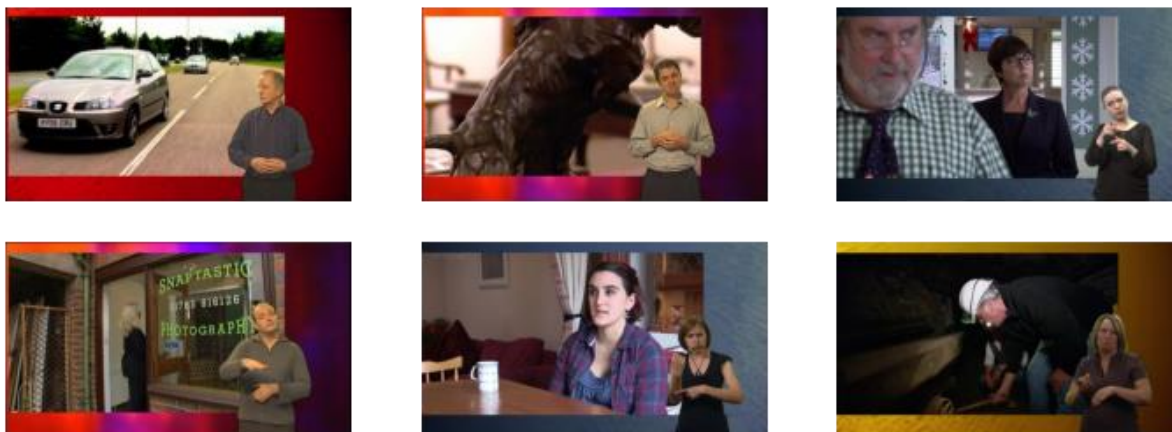


Figure 1: Sample images from BBC Pose and BBC Short Pose datasets

3.2 Image Preprocessing

The quality of the data used is one of the key aspects in the implementation of machine learning and deep learning. We found multiple factors in the BBC pose and the short BBC pose dataset that can potentially impact the performance of the prediction result, including missing human figures at the beginning and the ending of image frames (shown in Fig 2), Non-centralized characters (shown in Fig 3), background noise (shown in Fig 4), and unnormalized RGB values. Applying data preprocessing may lead to an enhancement in the accuracy of gender classification.

3.2.1 Missing Human Figure

Figure 2 demonstrates the missing human figure in the BBC pose dataset. Through observation, we found that the initial two to four minutes of a given video dataset could be the introduction, which does not contain any human figure. Filtering out these image outliers is vital before stepping into the joint predicting phase.

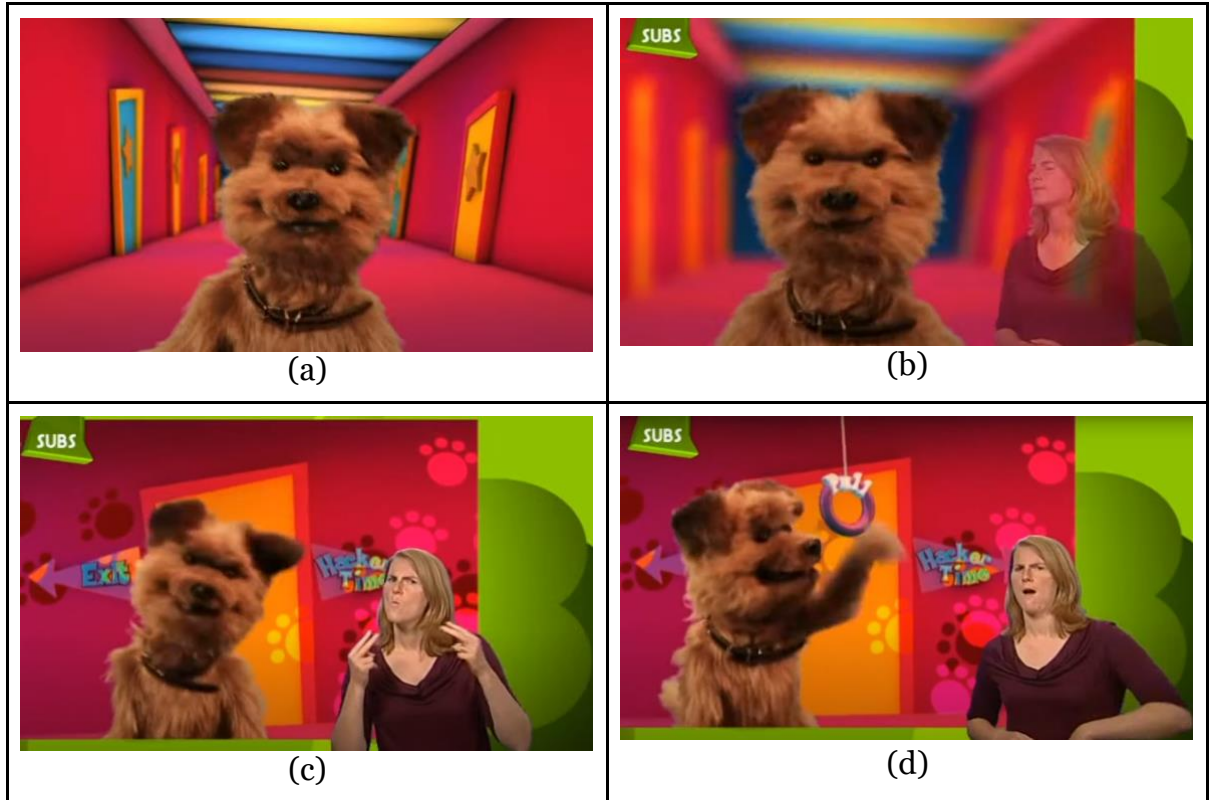


Figure 2: Sample image with missing human figure

- (a) Image frame with missing human figure*
- (b) image frame with an obscure human figure,*
- (c) (d) image frames with clear human figure*

3.2.2 Non-centralized character

Figure 3 shows the raw images that we extract from the BBC pose and short BBC pose dataset. By examining raw images from two data sources, we found that human figures commonly appear in the bottom right corner of the whole image frame. However, depending on different video datasets, some human figures show up a little bit closer to the center while others are located to the far right of the whole image. To get the

best joint prediction result, our goal is to centralize the human figure for each image frame with a fixed ratio. Since there are only fourteen videos, we manually determined the ratio and wrote a Python script to automatically crop the images per dataset with that fixed ratio.

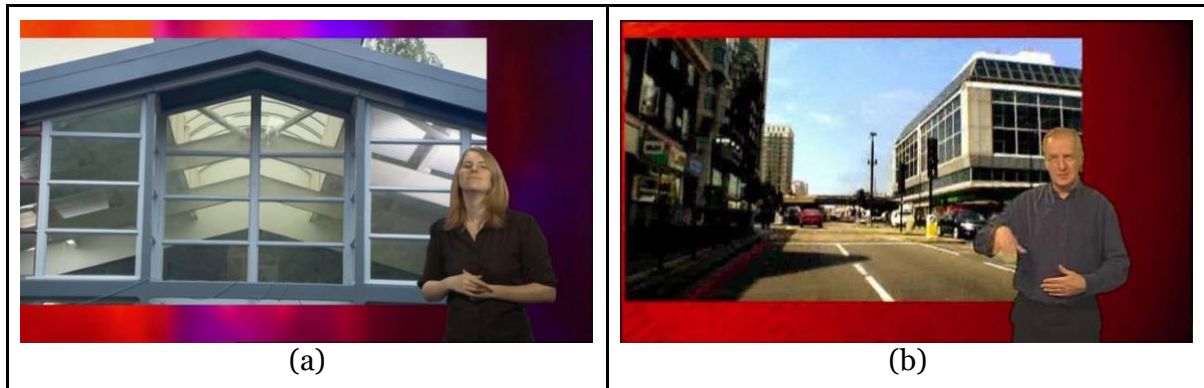


Figure 3: Non-centralized human figure

(a) Non-centralized female figure, (b) Non-centralized male figure

3.2.3 Background Noise

Another potential benefit of cropping images is to remove the background noise, which could largely increase the quality of joint prediction results especially when there are other human figures shown in the background image as demonstrated in Figure 4. Cropping out the background image so that we give our attention to the human figure at the bottom right corner of the whole image, allows us to get a better joint prediction result.



Figure 4: Background Noise

(a), (b) human figures showing in the background

Figure 5 presents the result of cropping images. For each video, we determined a fixed cropping ratio that centralizes the human figure while removing as much background noise as possible.

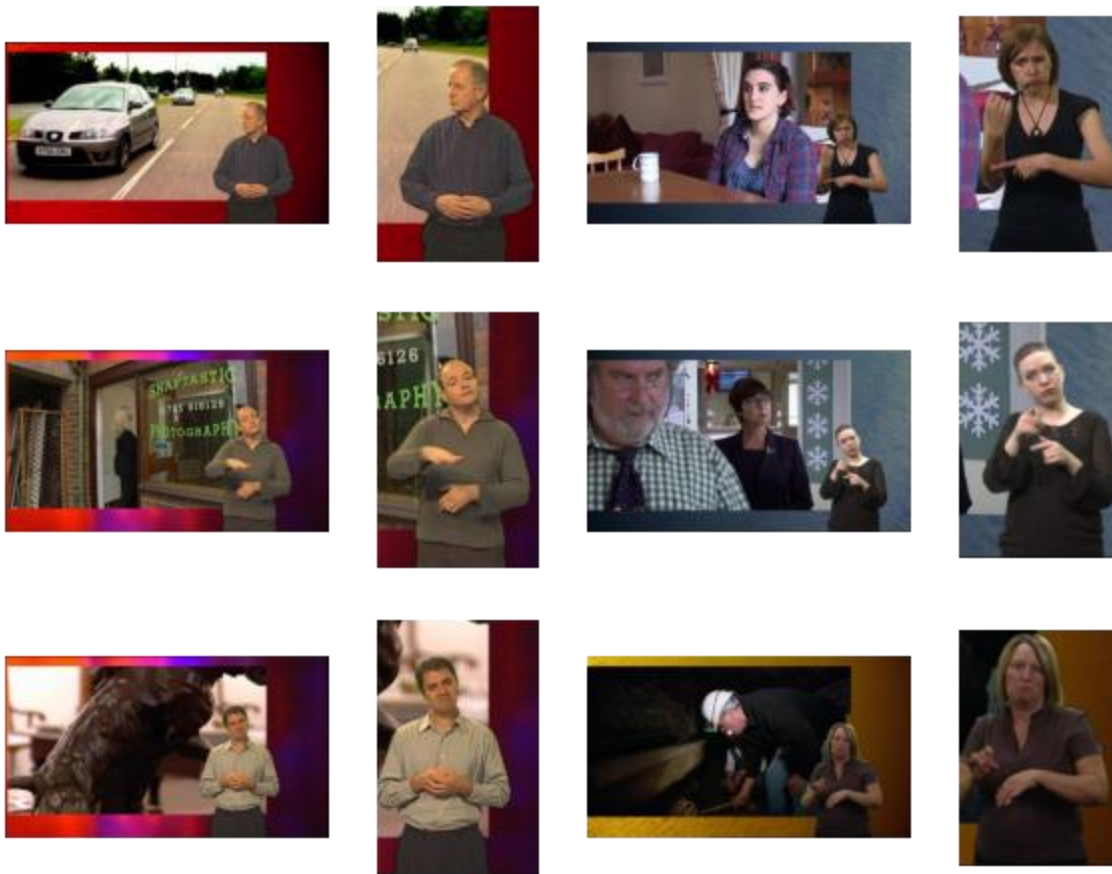


Figure 5: Cropping images with customized ratio

3.2.4 Joint Prediction

In this study, we adopted the method devised by B. Xiao et. al., [14] in 2018 for human joint predicting. B. Xiao et. al., [14] introduced a new neural network architecture, known as C5, by combining the ResNet with additional deconvolutional blocks appended at the end of the final convolution stage shown in Figure 6. The implementation of each deconvolutional block employs a deconvolutional layer, which has 256 filters with a 4x4 kernel followed up with a batch normalization and ReLU

activation function. Lastly, to generate the predicted heatmaps of k joints $\{H_1, H_2, \dots, H_k\}$, a 1×1 convolutional layer is added at the end.

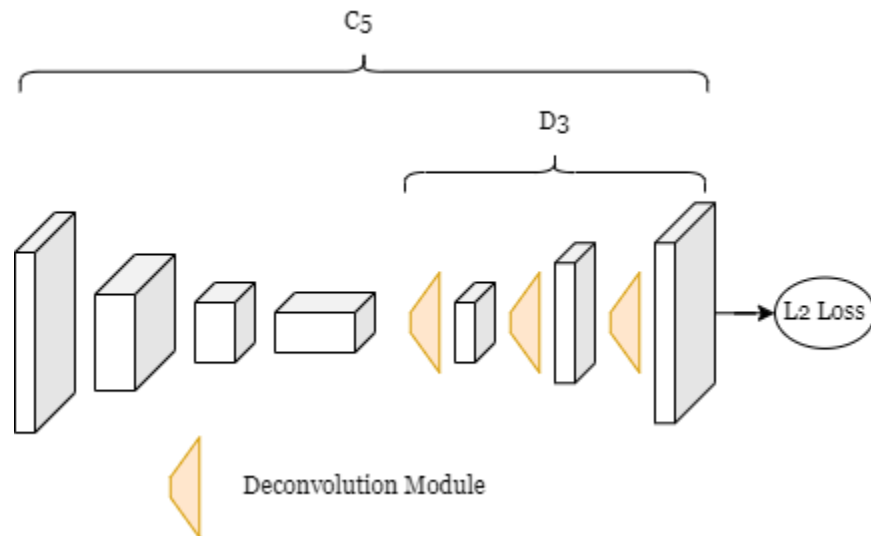


Figure 6: C5 Network architecture

In the C5 network, the input image is first resized into a 96-by-96 pixel picture and then processed by a series of convolutional and pooling layers to extract features, which are then fed into a fully connected layer that generates 2D heatmaps for each body joint such as the shoulders, elbows, hips, knees, etc. The top- k values in each heatmap represents the position of the corresponding body part in the image.

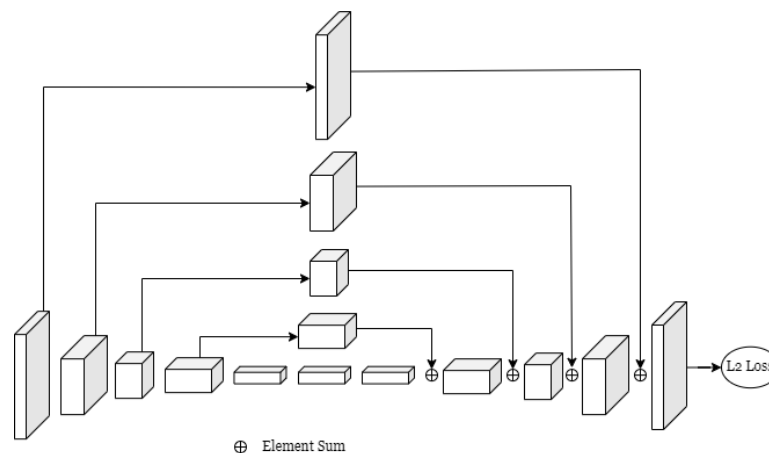


Figure 7: Hourglass Network architecture

Compared to some other state-of-the-art network architectures such as Hourglass [13] and CPN [15], which applied the upsampling with skip layer connections to enhance the feature map resolution as shown in Figures 7 and 8, C5, on the other hand, simply incorporate the upsampling and convolutional parameters into deconvolutional layers.

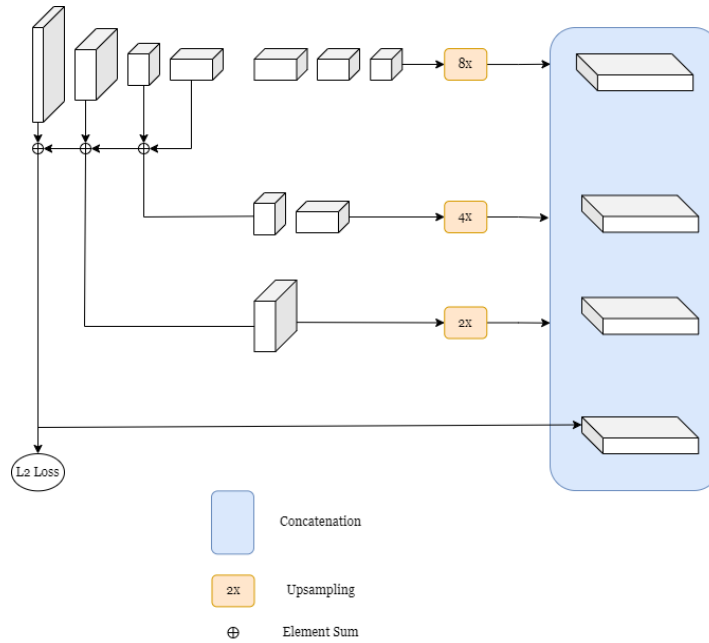


Figure 8: CPN Network architecture

Figure 9 demonstrates the outcome of sample images after undergoing all the preprocessing steps, which are listed as follows:

1. Image filtering (a)-(b): manually filters out image frames with missing human figure
2. Image cropping (a)-(b): centralize the character and reduce the background noise

3. Joint prediction (c): predicting joints via the heatmap regression model using the C5 network

The predicted joints (c) were then placed on top of the original image for evaluation and interpretation purposes

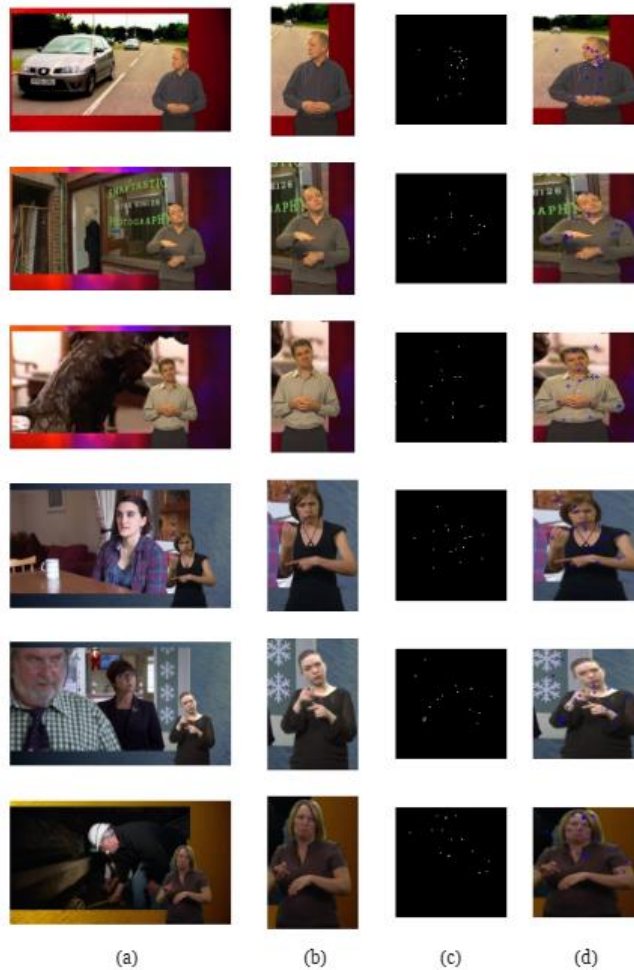


Figure 9: Image preprocessing with the image cropping and the joint prediction

- (a) Raw image frames sample from BBC Pose dataset,*
- (b) Image frames after cropping with customized ratio,*
- (c) Joint labeling result using C5 network,*
- (d) Predicted joints attached on the raw image frames*

IV. METHODOLOGY

In this section, we are going to introduce two methods that we proposed in this paper for predicting gender via joint images. For the classic machine learning method, we applied the Support Vector Machine (SVM) model. For deep learning methods, we built two Convolutional Neural Network (CNN) structures. We will introduce the SVM model in section 4.1 and before diving into the details configuration of the two CNNs structure in section 4.3, we will introduce the basic NN background section in 4.2.

4.1 SVM

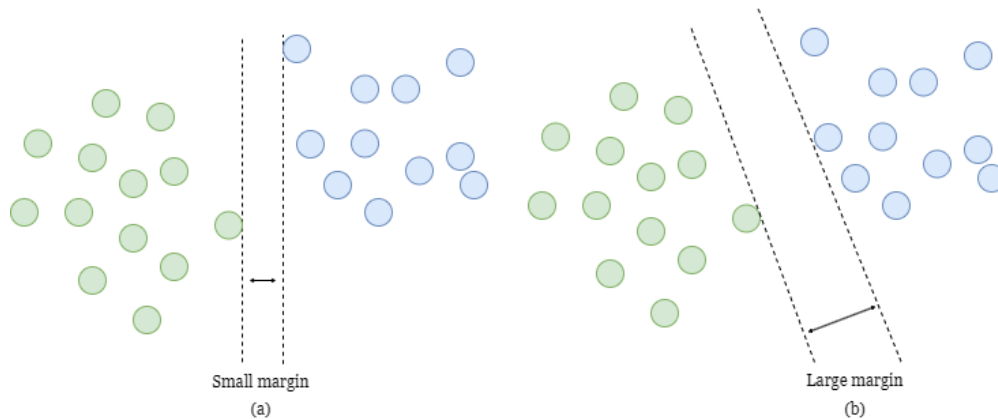


Figure 10: SVM

(a) SVM with a small margin,

(b) SVM with a large margin

Support Vector Machine (SVM), a supervised machine learning algorithm that is known for its ability to solve both classification and regression problems. For solving classification problems, it is commonly used for binary classification, where its goal is

to find an N-dimensional hyperplane. The hyperplane in the feature space can distinctly classify the data into two categories. There could be multiple hyperplanes exist that can separate two classes of data points. To measure how distinctness of the separation given a qualified hyperplane, the algorithm aims to optimize the maximum margin (i.e., the maximum distance) between two classes (shown in Figure 10). We train the SVM model by setting the maximum iteration of 2,500 iterations so that model will converge within a reasonable amount of time.

4.2 NN Background

4.2.1 ANN Model

A classic neural network structure typically includes three types of layers: input layer, hidden layer(s), and output layer.

- **Input layer:** the input layer is where we give our input data to our model. The number of neurons in the input layer is usually equivalent to the total number of features in given input data. In this study, image data is the input data, which consists of 96 times 96 one-channel pixels (i.e., grayscale). Thus, there are 9216 neurons in the input layer.
- **Hidden layer(s):** data that feed into the input layer is then forwarded to the hidden layer(s). As the name suggests, there could be more than one hidden layer depending on the data size and/or the design of the model. The output of each hidden layer is obtained by matrix multiplication of the previous output layer with trainable weights and biases. In order to make the network nonlinear,

the calculated result is then forwarded to an activation function such as ReLU, leaky ReLU (LReLU), and Hyperbolic tangent.

- Output layer: the output of the last hidden layer is then fed into a logistic function such as Sigmoid or SoftMax to come up with the probability value for each output class.

The error is then calculated for each output layer of the feedforward network via a loss function. To rectify these errors and minimize the total loss, backpropagation is applied by computing the gradient of the loss function along with the weights between each neuron connection.

4.2.2 CNN Model

The convolutional neural network, a special neural network structure for image recognition and tasks that involved pixel processing, is mainly used for retrieving neighboring spatial information. This type of network structure typically consists of convolutional layer(s), pooling layer(s), a fully connected (dense) layer, and a dropout layer.

4.2.2.1 Convolutional layers

Convolutional applied filters (or so-called kernels) to draw out features from the input data. Figure 11 shows how the convolution is computed by sliding the kernel from the top-left corner over the whole image data and performing an element-wise multiplication followed by a summation.

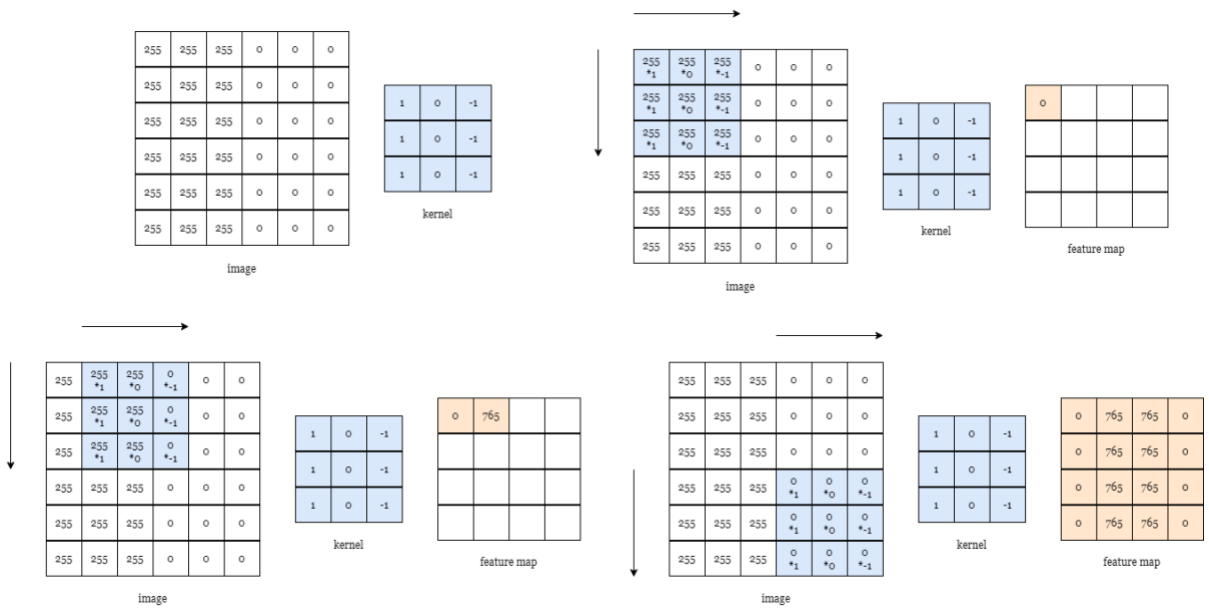


Figure 11: 3x3 Filter (Kernel)

Stride and padding are the other two parameters that can change the size and value of the output feature map. Stride denotes how many steps that kernel moves in each step in convolution. As the size of the stride increase, the resulting feature map will be smaller as shown in Figure 12.

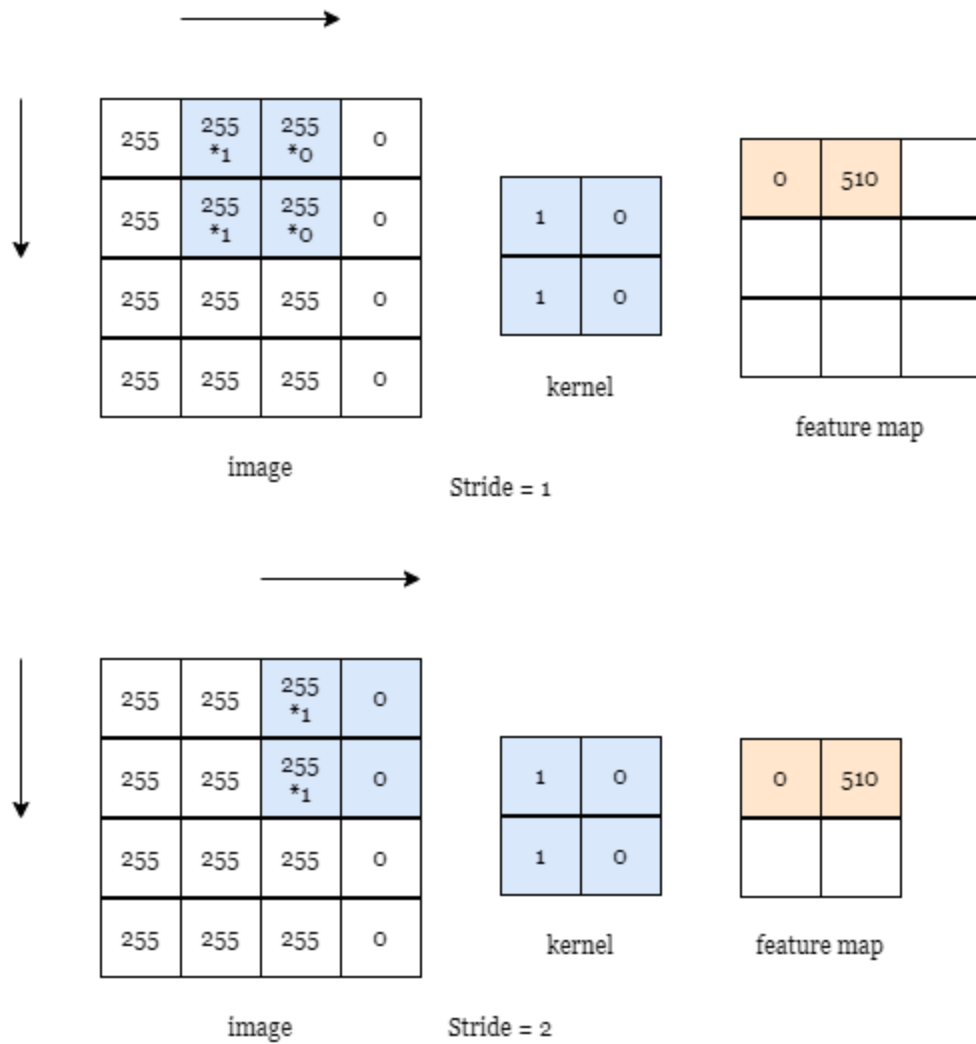


Figure 12: Stride

Padding is the technique of symmetrically adding values to the input matrix as shown in Figure 13. The main purpose of padding is to preserve the output dimension of the feature map by keeping it the same size as the input image.

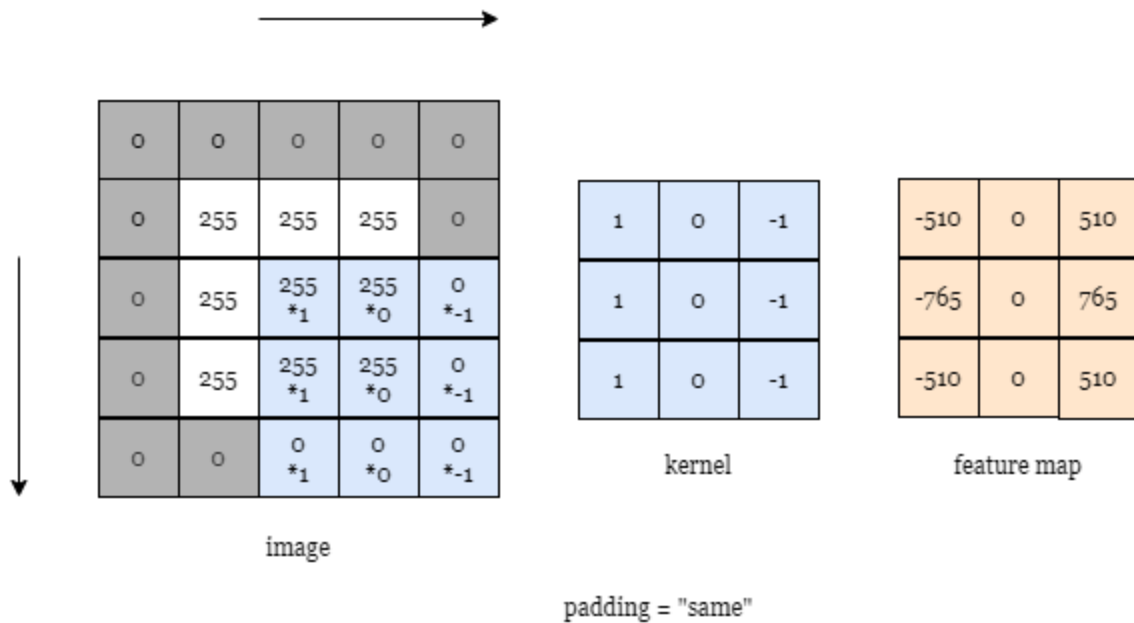


Figure 13: Padding

4.2.2.2 Pooling layer(s)

Pooling layer is the layer between two convolutional layers that is mainly used to reduce the size of the image width and height of the feature maps produced by the convolutional layer and subsample the spatial dimensions to capture the vital information. There are two types of pooling layers that are commonly used: Max pooling and Average pooling. Max pooling only consider the maximum value from each subregion of the feature map and passes it on to the next layer. Average pooling, on the other hand, takes the average of all the values in each subregion and passes it on to the next layer. Figure 14 presents the max pooling process of a 4 by 4 image to a 2 by 2 image.

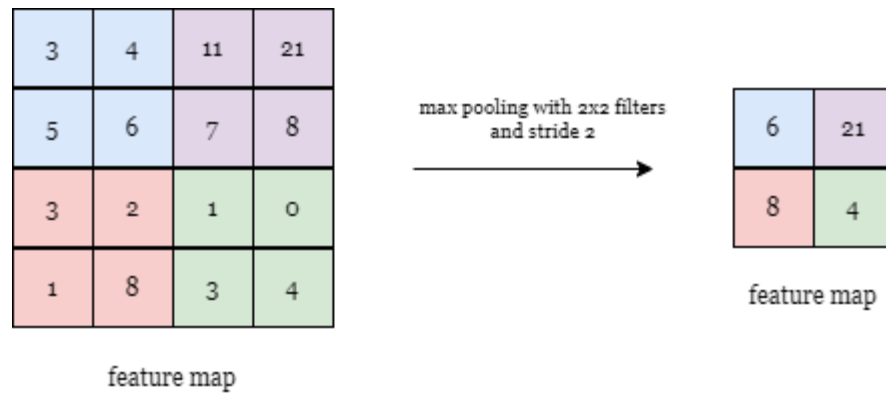


Figure 14: Pooling layer

4.2.2.3 Fully connected (Dense) layer

A fully connected layer is a layer that connects every neuron in the previous layer. The output of the fully connected layer for the classification problem is a vector with scores between 0 to 1 inclusive to represent the probability for each prediction class. Figure 15 shows the network structure for the fully connected layer

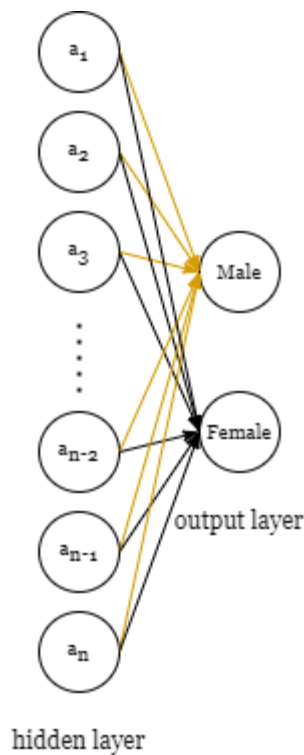


Figure 15: Fully connected (Dense) layer

4.2.2.4 Dropout layer

Dropout is a regularization technique that has been believed to prevent overfitting. Overfitting refers to the training model that learns too well to the training data while having poor performance to the generally unseen data. Dropout is implemented as a layer of a neural network to randomly drop out a certain fraction of neurons in the previous layer during each feedforward process. The dropout rate, which determines the fraction of activation to be dropped is a hyperparameter of the neural network model that is typically set to a value between 0.5 to 0.8. Figure 16 presents a hidden dropout layer in a toy neural network structure.

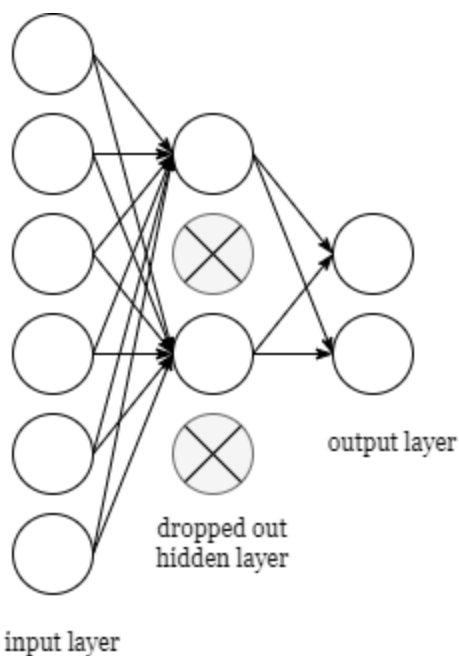


Figure 16: Dropout layer

4.3 Building CNN model

Within the scope of our study, we built two different CNN structures for gender classification: shallow and deep CNN structures. The convolutional layers were followed by a dropout layer, a ReLU activation layer, and a max pooling layer (enclosed in a green rectangle in Figures 17 and 18). The detailed breakdown of two convolutional neural network architectures is shown below in Tables 2 and 3. For both models, we set the loss to the sparse categorical cross-entropy and applied the Adam algorithm as our model optimizer.

4.3.1 Shallow (1-layered) CNN



Figure 17: Shallow CNN Structure

Table 2: Shallow CNN Architecture

Layer	Output shape	Param #
input	(96, 96, 1 / 3)	0
conv2d	(96, 96, 32)	320 / 896
dropout	(96, 96, 32)	0
activation (ReLU)	(96, 96, 32)	0
max_pooling	(48, 48, 32)	0
flatten	(73728)	0
dense	(128)	9437312
dropout	(128)	0
dense	(2)	258

4.3.2 Deep (4-layered) CNN

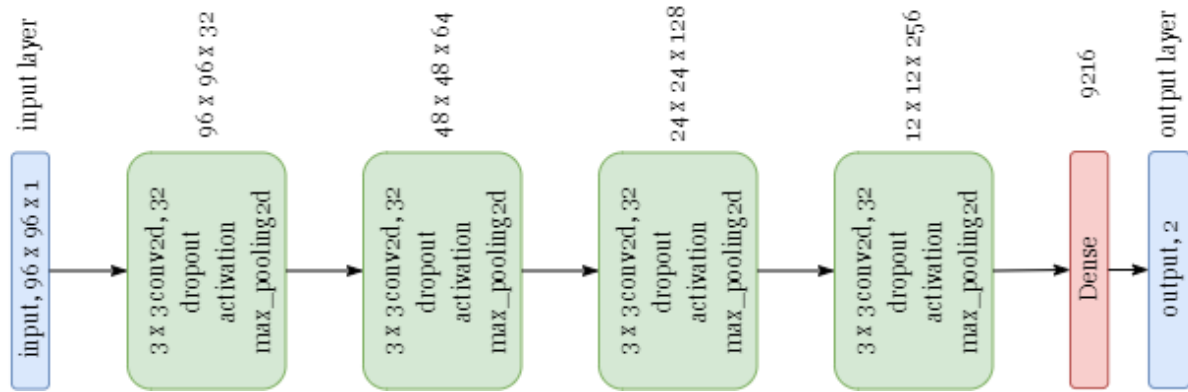


Figure 18: Deep CNN Structure

Table 3: Deep CNN Architecture

Layer	Output shape	Param #
input	(96, 96, 1/3)	0
conv2d	(96, 96, 32)	320 / 896
dropout	(96, 96, 32)	0
activation (ReLU)	(96, 96, 32)	0
max_pooling	(48, 48, 32)	0
conv2d	(48, 48, 64)	18496
dropout	(48, 48, 64)	0
activation (ReLU)	(48, 48, 64)	0
max_pooling	(24, 24, 64)	0
conv2d	(24, 24, 128)	73856
dropout	(24, 24, 128)	0
activation (ReLU)	(24, 24, 128)	0
max_pooling	(12, 12, 128)	0

conv2d	(12, 12, 256)	320
dropout	(12, 12, 256)	0
activation (ReLU)	(12, 12, 256)	0
max_pooling	(6, 6, 256)	0
flatten	(9216)	0
dense	(128)	1179776
dropout	(128)	0
dense	(2)	258

V. EXPERIMENT

Table 4: Snapshot of training and validation results on epochs 10, 20 and 30

Algorithm	Input Image	Epochs	Training Loss	Validation Loss	Training Accuracy	Validation Accuracy
Shallow CNN	Joint	10	0.0456	2.6947	0.9824	0.6086
		20	0.0273	3.9406	0.9908	0.6148
		30	0.0206	4.6845	0.9928	0.6212
Shallow CNN	Joint + RGB Image	10	0.0061	0.9878	0.9998	0.9583
		20	0.0190	2.7593	0.9999	0.9423
		30	0.0056	1.1672	0.9997	0.9962
Deep CNN	Joint	10	0.2002	1.2442	0.9380	0.6446
		20	0.1570	1.3997	0.9550	0.6177
		30	0.1385	1.1087	0.9630	0.6731
Deep CNN	Joint + RGB Image	10	0.1227	2.0610	0.9998	0.9954
		20	0.0429	0.5786	0.9994	0.7745
		30	0.0229	0.5405	0.9984	0.8366

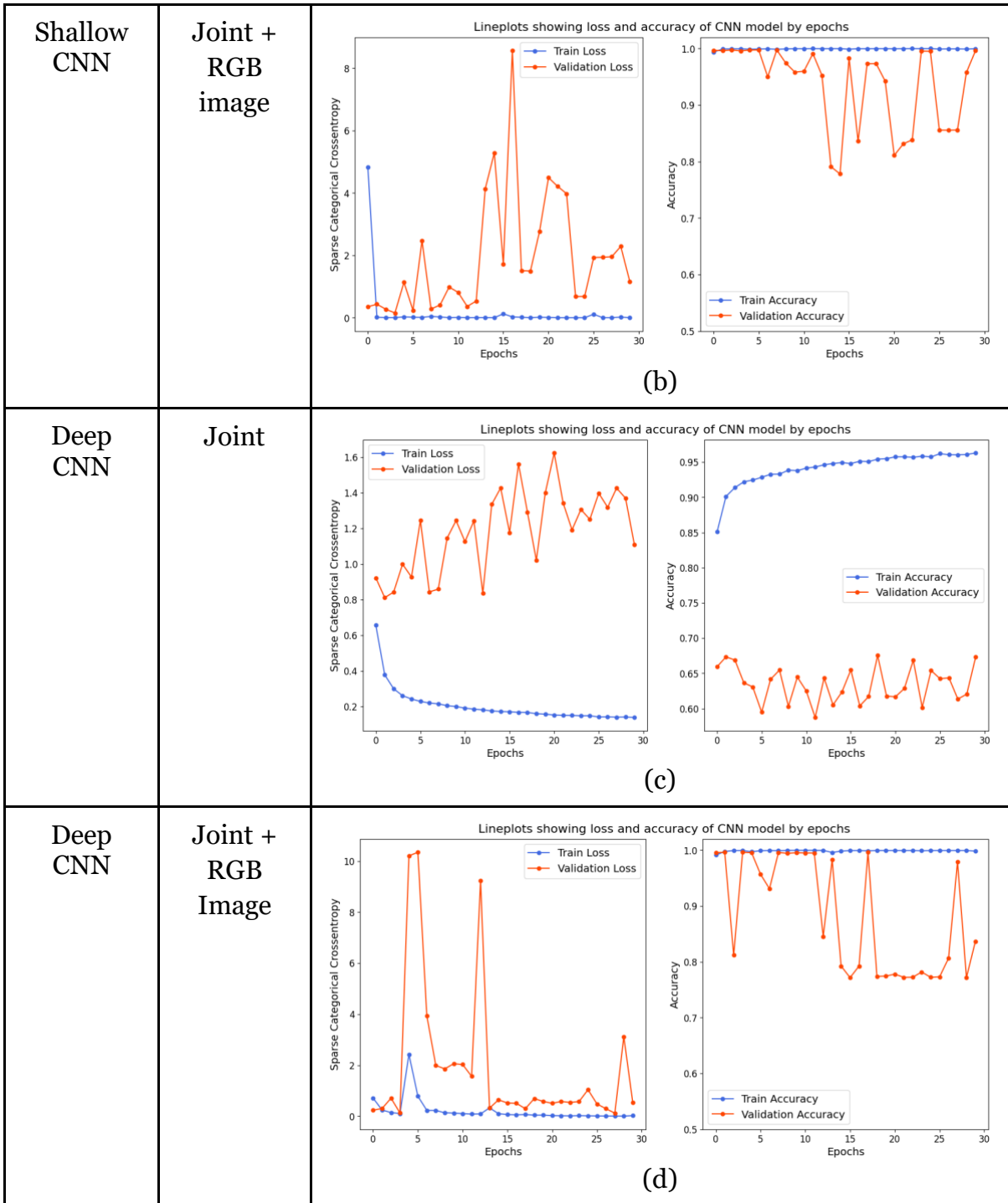
Tables 3 and 4 presents the detail training and validation phases on two CNN networks at every interval of 10 epochs. Both CNN architectures were trained on two kinds of

image input, pure joint-label images, and joint labels attached to the raw RGB images. Comparing two network structures (shallow and deep CNNs) that trained on purely joint images, we noticed that the shallow CNN shows higher training accuracy and lower training loss as the number of training epochs increased. However, the higher validation loss that steadily and quickly rose on shallow CNN suggests that the model is suffering from severe overfitting issues compared to the Deep CNN structure. The superiority of the deep CNN structure compared to the shallow CNN structure was attested in Table 5, where testing accuracy on Deep CNN outperformed shallow CNN by about 5%.

Table 5: Loss and accuracy in figure format

- (a) Training/validation loss/acc on Joint images using shallow CNN
- (b) Training/validation loss/acc on Joint label + raw images using shallow CNN
- (c) Training/validation loss/acc on Joint images using deep CNN
- (d) Training/validation loss/acc on Joint label + raw images using deep CNN

Algorithm	Input format	Loss and Accuracy
Shallow CNN	Joint	<p style="text-align: center;">(a)</p>



The training process of both network structures on RGB images with joint labels attached was also shown in tables 3 and 4. As we can tell from the tables above, both

models obtained similar training and validation accuracy for most of the training epochs. Both models received comparable testing accuracy of around 99.6% classification accuracy as shown in table 5.

Table 6: Testing Result (accuracy, recall precision, and F1 score) on four models

Algorithm	Input format	Accuracy	Recall	Precision	F1 score
SVM	Joints	0.563	0.563	0.585	0.557
Shallow CNN	Joints	0.621	0.621	0.646	0.616
Shallow CNN	Joints + RGB Image	0.996	0.996	0.996	0.996
Deep CNN	Joints	0.665	0.665	0.673	0.664
Deep CNN	Joints + RGB Image	0.996	0.996	0.996	0.996

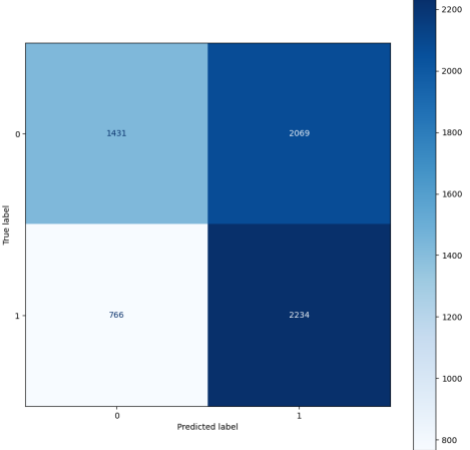
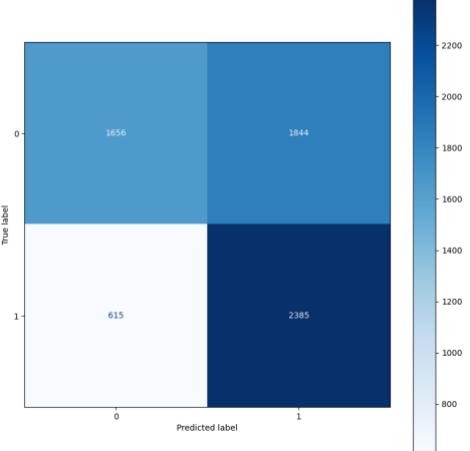
Table 6 displayed the accuracy, recall, precision, and F1 score of SVM and four models that are composed of different image sources and network architecture. Deep CNN structure surpasses the SVM model and the shallow CNN in all kinds of evaluation metrics when predicting gender solely relying on the joint labels. From table 6, we observe that both deep and shallow CNN perform extremely well (about 99.6% testing accuracy) when predicting gender along with the joints and the hint of raw images. This further demonstrating the difficulty of solving the problem of this project that input only with the coordinates of 2-D body joints. Compared our proposed CNN

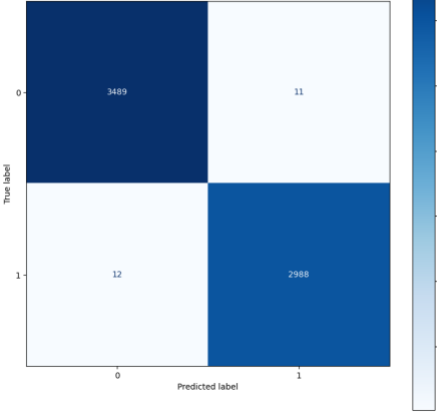
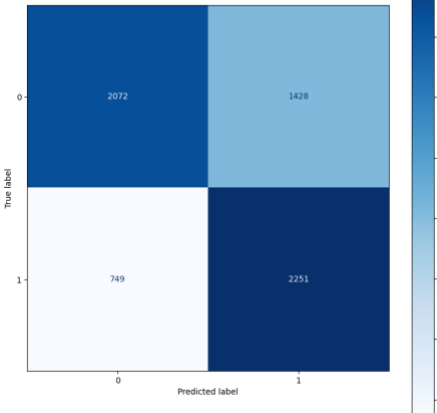
structures (shallow and deep CNN) with the classic machine learning model, the deep CNN outperform SVM about 10% under the same input format of joint images. From tables 6 and 7, it is worth noting that the majority of the misclassification (False Positive, FP, and False Negative, FN) for both models that trained on purely joint labels seems to be class 1, which indicates the male gender. The reason for that could be attributed to the limited number of videos featuring female subjects in the BBC video dataset. To ensure our models train on a sufficient number of training data, we supplement each female video dataset with two male video datasets while sampling only half of the image frames from male acting videos in comparison to female acting videos.

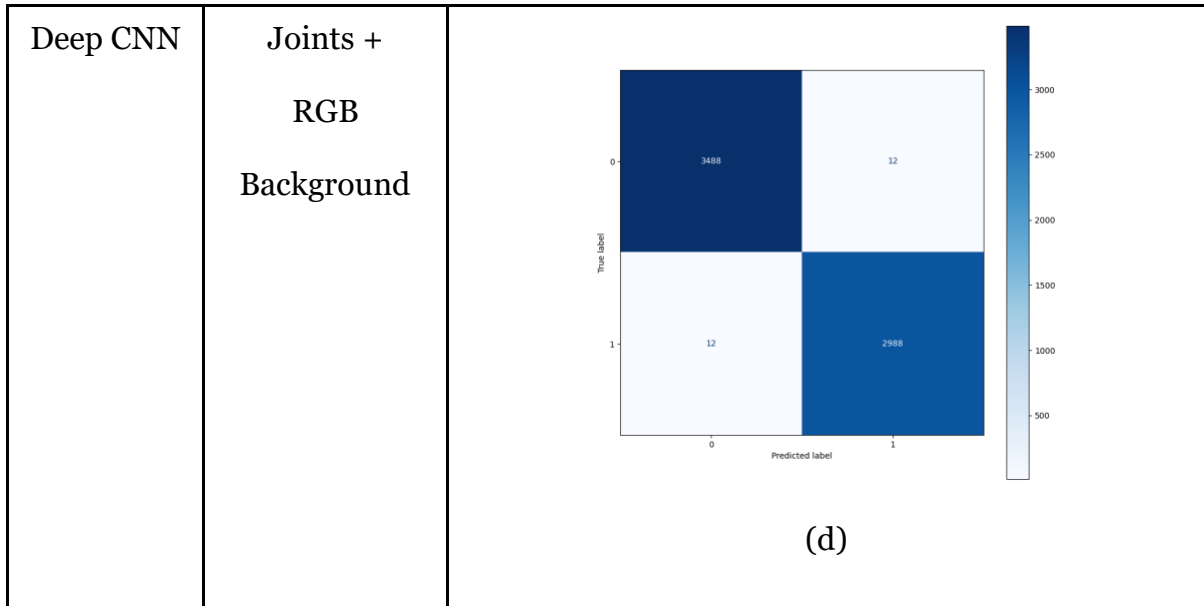
Table 7: Confusion Matrix

- (a) *Confusion matrix on Joint images using SVM,*
- (b) *Confusion matrix on Joint images using shallow CNN,*
- (c) *Confusion matrix on Joint label + raw images using shallow CNN,*
- (d) *Confusion matrix on Joint images using deep CNN,*
- (e) *Confusion matrix on Joint label + raw images using deep CNN*

Algorithm	Input format	Confusion Matrix
-----------	--------------	------------------

<p>SVM</p>	<p>Joints</p>	 <table border="1" data-bbox="824 262 1291 709"> <thead> <tr> <th>True label \ Predicted label</th> <th>0</th> <th>1</th> </tr> </thead> <tbody> <tr> <th>0</th> <td>1431</td> <td>2069</td> </tr> <tr> <th>1</th> <td>766</td> <td>2234</td> </tr> </tbody> </table>	True label \ Predicted label	0	1	0	1431	2069	1	766	2234
True label \ Predicted label	0	1									
0	1431	2069									
1	766	2234									
<p>Shallow CNN</p>	<p>Joints</p>	 <table border="1" data-bbox="824 751 1291 1199"> <thead> <tr> <th>True label \ Predicted label</th> <th>0</th> <th>1</th> </tr> </thead> <tbody> <tr> <th>0</th> <td>1656</td> <td>1844</td> </tr> <tr> <th>1</th> <td>615</td> <td>2385</td> </tr> </tbody> </table> <p>(a)</p>	True label \ Predicted label	0	1	0	1656	1844	1	615	2385
True label \ Predicted label	0	1									
0	1656	1844									
1	615	2385									

<p>Shallow CNN</p>	<p>Joints + RGB Background</p>	 <p>(b)</p>
<p>Deep CNN</p>	<p>Joints</p>	 <p>(c)</p>



VI. DISCUSSION

While our study showed promising results on classifying the gender simply using a joint image of a human-figure, we acknowledge that the majority of individuals in the BBC pose, and short BBC pose dataset limited to Western people. Therefore, it is vital to comprehend the outcome of our research result with caution and knowing that the result might not generalize to other populations or datasets. We encourage further research on the same topic but with different datasets to fully understand the whole picture. Nevertheless, we still believe that our study of this project provides a valuable beginning.

VII. CONCLUSION AND FUTURE WORK

In this study, we adopted a human joints pre-trained model, named C5, to label human gestures for gender classification using the Convolution Neural Network models. To further improve the quality of the labeling result, we preprocessed the input images via cropping and normalizing through mean and standard deviation that is used by ImageNet pre-trained model, which result in 66.5% accuracy. Future works include increasing the training images by adding more frames from various videos of the BBC human pose estimation dataset and/or performing OpenCV techniques for data augmenting such as ImageDataGenerator by adjusting the angle (via rotation) and brightness of the training images. Besides, data cleaning also belongs as a critical factor in the quality of labeling results as occasionally, the background of the training image contains random noise, which leads to the wrong joints-labeling result. Last but not least, adopting another pose-estimation model could possibly result in better prediction accuracy especially when the pre-trained model is trained on the BBC dataset.

REFERENCE

- [1] AM. Burton, V. Bruce, N. Dench. What's the difference between men and women? Evidence from facial measurement. *Perception*. 1993;22(2):153-76. doi: 10.1068/p220153. PMID: 8474841.
- [2] AJ. O'Toole, KA. Deffenbacher, D. Valentin, K. McKee, D. Huff, H. Abdi "The perception of face gender : The role of stimulus structure in recognition and classification," vol. 26, no. 1, pp. 146–160, 1998, doi: 10.3758/BF03211378.
- [3] MK. Scheuerman, M. Pape, and A. Hanna, "Auto-essentialization: Gender in automated facial analysis as extended colonial project," vol. 8, no. 2, p. 205395172110537, 2021, doi: 10.1177/20539517211053712.
- [4] SS. LIEW, M. KHALIL-HANI, S. AHMAD RADZI, and R. BAKHTERI, "Gender classification: a convolutional neural network approach," vol. 24, pp. 1248–1264, 2016, doi: 10.3906/elk-1311-58.
- [5] J. Mazurkiewicz, "Gender Recognition System Based on Human Face Picture." *Journal of Polish Safety and Reliability Association*. 2017, 8, (1), s. 97--104.
- [6] MS. Fathollahi, R. Heidari, "Gender classification from face images using central difference convolutional networks," *Int. J. Multim. Inf. Retr.*, vol. 11, no. 4, pp. 695–703, 2022. [Online]. Available: <https://doi.org/10.1007/s13735-022-00259-0>

- [7] A. Lahariya, V. Singh, US. Tiwary, “Real-time emotion and gender classification using ensemble CNN,” CoRR, vol. abs/2111.07746, 2021. [Online]. Available: <https://arxiv.org/abs/2111.07746>
- [8] O. Arriaga, M. Valdenegro-Toro, P. Piöoger, “Real-time convolutional neural networks for emotion and gender classification,” CoRR, vol. abs/1710.07557, 2017. [Online]. Available: <http://arxiv.org/abs/1710.07557>
- [9] C. Xu, Y. Makihara, R. Liao, H. Niitsuma, X. Li, Y. Yagi, J. Lu, “Real-time gait-based age estimation and gender classification from a single image,” in IEEE Winter Conference on Applications of Computer Vision, WACV 2021, Waikoloa, HI, USA, January 3-8, 2021. IEEE, 2021, pp. 3459–3469. [Online]. Available: <https://doi.org/10.1109/WACV48630.2021.00350>
- [10] M. Azhar, S. Ullah, K. Ullah, I. Syed, and J. Choi, “A gait-based real-time gender classification system using whole body joints,” Sensors, vol. 22, no. 23, p. 9113, 2022. [Online]. Available: <https://doi.org/10.3390/s22239113>
- [11] J. Charles, T. Pfister, M. Everingham, A. Zisserman, “Automatic and Efficient Human Pose Estimation for Sign Language Videos,” vol. 110, no. 1, pp. 70–90, 2014, doi: 10.1007/s11263-013-0672-6.
- [12] J. Charles, T. Pfister, D. Magee, D. Hogg, A. Zisserman, “Domain adaptation for upper body pose tracking in signed TV broadcasts,” in British Machine Vision Conference, 2013

- [13] A. Newell, K. Yang, J. Deng, “Stacked hourglass networks for human pose estimation.” In: European Conference on Computer Vision. pp. 483–499. Springer (2016)
- [14] B. Xiao, H. Wu, Y. Wei, “Simple Baselines for Human Pose Estimation and Tracking.” 2018, doi: 10.48550/arxiv.1804.06208.
- [15] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, J. Sun, “Cascaded pyramid network for multi-person pose estimation.” In: CVPR (2018)