CHAPTER II. LITERATURE STUDY

2.1 Literature Study

Some of the studies, research and journal that are used as references in this study are as follows:

Based on research (Deng et al., 2019) entitled "RetinaFace: Single-stage Dense Face Localization in the Wild", RetinaFace produces the best AP in all subsets of both validation and test sets, i.e., 96.9% for Easy validation set, 96.1% for Medium validation set and 91.8% for Hard validation set, and 96.3% for Easy test set, 95.6% for Medium test set and 91.4% for Hard test set. Compared to the latest and the best performed method, RetinaFace surpass it with set a new impressive record (91.4% vs. 90.3%) on the Hard subset (image test) which contains a large number of tiny faces in single image.

Based on research (Deng et al., 2020.) entitled "RetinaFace: Single-shot Multi-level Face Localisation in the Wild", RetinaFace-MobileNet0.5 surpass the baselines and decrease the failure rate up to 19.72%. By using a better backbone (ResNet-50), RetinaFace-R50 reduces the failure rate up to 9.82%. After removing the 3D mesh regression branch from RetinaFace, the AUC decreases significantly from 58.54% to 55.66%. This is because 3D mesh regression is pose in variant and a joint training framework can improve the accuracy of 2D five facial landmarks.

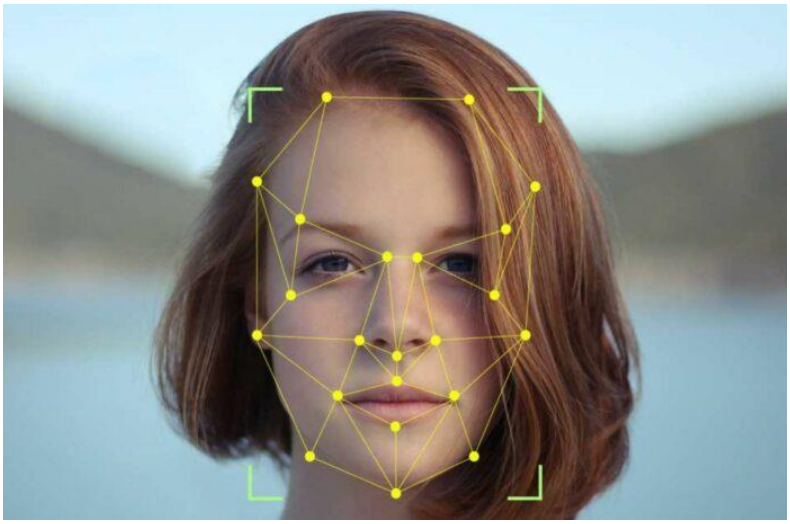
Based on research (F. Zhang et al., 2019) entitled "Accurate Face Detection for High Performance", present a high performance face detector by equipping the popular one-stage RetinaNet method with some recent tricks: (1) Employing the two-step classification and regression for detection; (2) Applying the Intersection over Union (IoU) loss function for regression; (3) Revisiting the data augmentation based on data-anchor-sampling for training; (4) Utilizing the max-out operation for robust classification; (5) Using the multi-scale testing strategy for inference.

2.2 Basic Theory

2.2.1. Facial Recognition

Facial recognition is a way of recognizing a human face through technology. As the information on human identity, human faces are unique, and the face recognition system uses faceprint values as a feature from a photograph or video(Rahmad et al., 2021). Facial recognition is very different from the conventional camera surveillance / CCTV, it is not just a recording some selected areas, but rather it does identification of an individual images or faces

by comparing most recent capture images or faces with those images that already saved in a database or trained images in models.

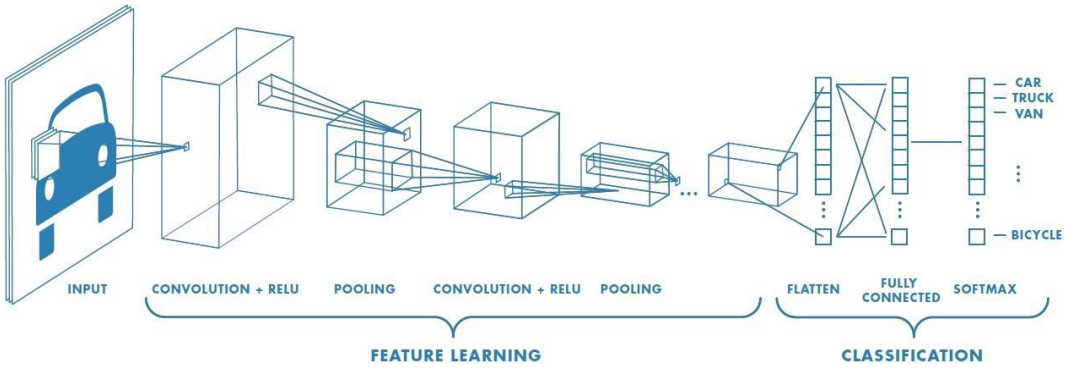


Figures 2.1 Face Recognition (illustration)

Source: (Belen Martin, 2020)

2.2.2. Convolutional Neural Netwotk (CNN)

CNN / Convnet is a Deep Learning algorithm, where algorithm can take an input image or computer vision and assign it to various aspects in the image and we can differentiate one image to another.



Figures 2.2 CNN Sequence (illustration)

Source: (Sumit Saha, 2018)

The preprocessing that required in CNN is much lower that other method. And this method is the method very often used in image recognition systems. There are various architectures of CNN that available which become a key in building algorithm in Image recognition. Some of them are: 1) LeNet, 2) AlexNet, 3) VGGNet, 4) GoogleNet, 5)ResNet, 6)ZFNet, 7) ArcFace.

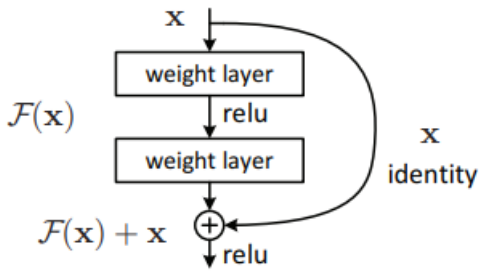
CNN works with preprocessing the inputted image with separated their primary color (Red, Green , Blue), the role of CNN will convert the image into the form which is easier to

process without losing their features and we can get a very good prediction later on, and then the CNN used the Kernel / Filter K to carrying the involved element for convolution operation, the purpose of Convolutional Operation is to extract the high-level features, from input images. And then there are process named Pooling layer, Pooling layer works for reducing the spatial size of the Convolved features, and the last process is Classification - Fully Connected Layer (FC Layer), in this process CNN make a linear combinations of High Level features as represented by the convolutional layer, The Fully Connected Layer is learning possibly non-linear function, and CNN convert our input image that appropriate for recognition(Sumit Saha, 2018).

2.2.3. Residual Neural Network (ResNet)

Deep Residual Network or Resnet is one of the best architectures in CNN. This architecture was created to solved deeper / harder neural network (adding more layers in neural network) or Deep Learning to solve complicated task and to improve classification or recognition accuracy. ResNet can effeciently train 100 layers and 1000 layers also without reducing any performance.

ResNet model use the skip connection on two or three layer using ReLu and batch normalization in between in their architecture, this can be called residual block.(He et al., 2015)



Figures 2.3 Residual Block (illustration)

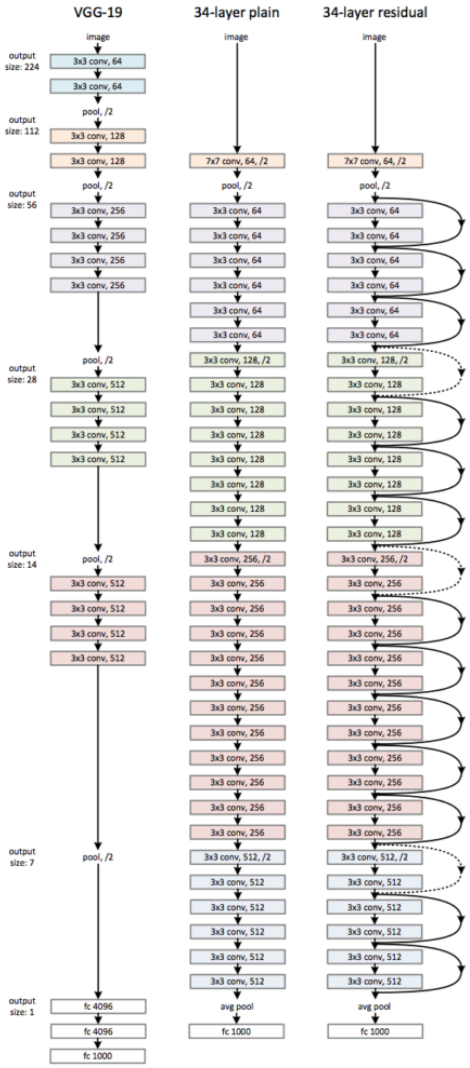
Source: (He et al., 2015)

Block in Residual network can be defined as

$$\mathbf{y} = \mathcal{F}(\mathbf{x}, \{W_i\}) + \mathbf{x}$$

x and **y** are the input and output vectors of the layers. The function $\mathcal{F}(\mathbf{x}, \{W_i\})$ represents the residual mapping to be learned (He et al., 2015).

The skip connection in ResNet helps to solve the problem of gradient that disapear in deep neural network. By permitting this alternate path for gradient to flow through. ResNet network uses a 34-layer plain network architecture inspired by VGG-19.



Figures 2.4 VGG-19 and 34 Layer (illustration)

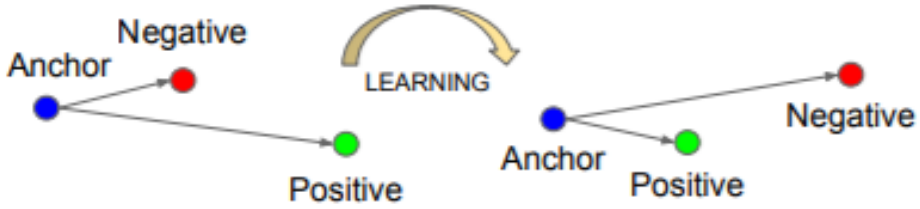
Source: (He et al., 2015)

ResNet has several distinct types of architecture based on the number of layers used, ranging from 18 layers, 34 layers, 50 layers, 101 layers, up to 152 layers(He et al., 2015). And Keras Applications have the following ResNet best practice and provide ResNet V1 and ResNet V2 with 50, 101, or 152 layers.

2.2.4. FaceNet

FaceNet is a Face Recognition system developed in 2015 that achieved the state-of-art result on a face recognition benchmark dataset, the FaceNet system can be used to extract high-quality features from faces that provided in images, called face embeddings, that can then be used to train a face identification system. FaceNet that directly learns a mapping from face images to a compact Euclidian space, where distances directly correspond to a measure of face similarity (Schroff et al., 2015).

The difference between FaceNet and other method, is that FaceNet learns the mapping from the images or faces and creates embeddings rather than using any bottleneck layer for recognition or verification tasks. Once the embeddings are created all the other tasks like verification and recognition can be performed using standard techniques (Schroff et al., 2015), FaceNet use Triplet loss, that will directly reflects what we want to achieve in face verification, recognition and clustering(Schroff et al., 2015). Triplet Loss function is to get our anchor image (specific image of person A) to get closer with image positive (Images that contain person A only) and compared to negative image (Other images with contain another person). So, we can get better recognition with comparison from positive and negative image.



Figures 2.5 Triplet Loss (illustration)

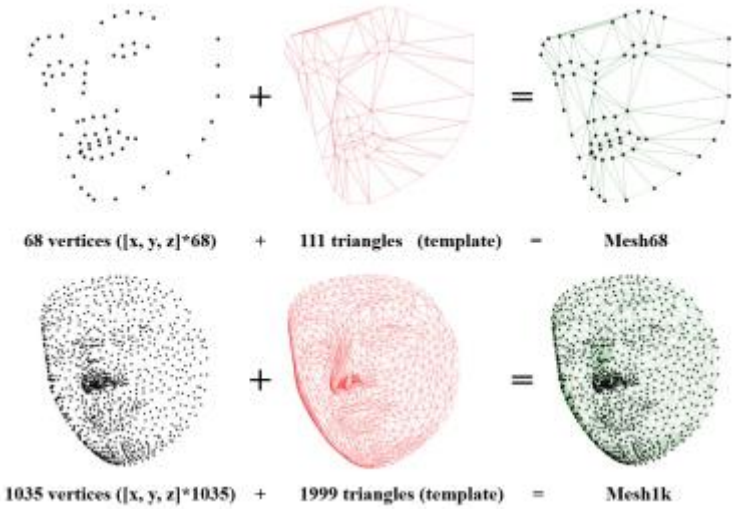
Source: (Schroff et al., 2015)

2.2.5. RetinaFace

Pixel-wise face localization method, which utilize a multi-task learning strategy to simultaneously predict face score, face box, five facial landmarks, and 3D position and correspondence of each facial pixel (Deng et al., 2019). RetinaFace surpass the accuracy point of state of the art two-stage method, RetinaFace can improve ArcFace's verification accuracy (with TAR equal to 89.59% when FAR=1e-6). This indicates that better face localization can significantly improve face recognition (Deng et al., 2019).

2.2.6. 3D Face Reconstruction

For creating a 3D face from the 2D image, predefined triangular face with N vertices as shown in the above figure, the vertices shares the same semantic meaning across different faces and with the fixed triangular topology each face pixel.



Figures 2.6 3D Face Reconstruction (illustration)

Source: (Deng et al., 2020.)

3D vertices on the 2D image plane, using 2 loss functions:

$$\mathcal{L}_{\text{vert}} = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{V}_i(x,y,z) - \mathbf{V}_i^*(x,y,z)\|_1$$

N is the total vertices i.e. 1103(68+1035) and $\mathbf{V}_i$ is predicted point and $\mathbf{V}_i^*$ is ground-truth point. The x and y coordinates of visible vertices in the image space can be directly learned from input face images.

$$\mathcal{L}_{edge} = \frac{1}{3M}\sum_{i=1}^{M} \|\mathbf{E}_i - \mathbf{E}_i^*\|_1$$

It is the edge length loss, as it is a triangular topology. Here, M is the number of triangles i.e., 2110(111+1999) and is $\mathbf{E}_i$ is predicted edge length and $\mathbf{E}_i^*$ is ground truth edge length.

By combining the vertex loss ($\mathcal{L}_{\text{vert}}$) and the edge loss ($\mathcal{L}_{\text{edge}}$), we define the mesh regression loss ($\mathcal{L}_{\text{mesh}}$) as:

$$\mathcal{L}_{\text{mesh}} = \mathcal{L}_{\text{vert}} + \lambda_0 \mathcal{L}_{\text{edge}}$$

where λ_0 is set to 1 according to our experimental experience.

2.2.7. Single-shot Multi-level Face Localization

The model consists of three main components:

a) Feature Pyramid Network

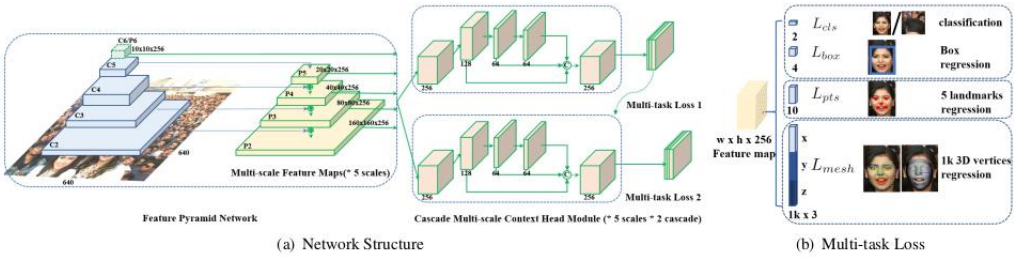
It makes input image and outputs five feature maps of different scales (Mohd Nayeem, 2020).

b) Context Head Module

To increase accuracy of context modelling capacity deformation convolutional network (DCN) (Mohd Nayeem, 2020).

c) Cascade Multitask Loss

To improve face localization cascade regression along with multi-task loss (Mohd Nayeem, 2020).



Figures 2.7 Single-shot multi-level face localization (illustration)

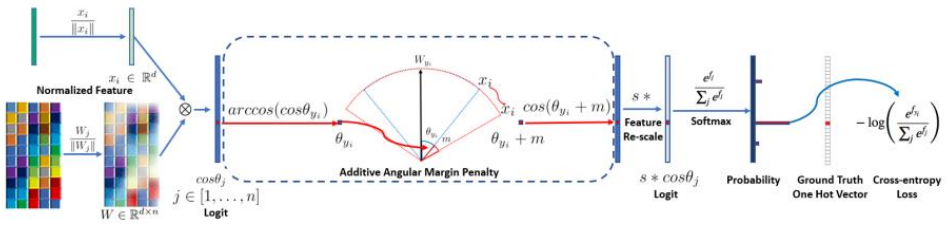
Source: (Deng et al., 2020.)

2.2.8. ArcFace (Additive Angular Margin Loss)

Is a Loss function used in face recognition task, the SoftMax is traditionally used in this task, ArcFace can be used to obtain highly discriminative features for face recognition, obtain highly discriminative features for face recognition, The proposed ArcFace has a clear geometric interpretation due to the exact correspondence to the geodesic distance on the hypersphere (Deng et al., 2018). The most widely used classification is SoftMax loss, presented as follows:

$$L_1 = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{W_{yi}^T x_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_j^T x_i + b_j}}$$

where xi ∈ R d indicate the deep feature of the i-th sample, belongs to the yi-th class(Deng et al., 2018). The embedding feature dimension d is set to 512, Wj ∈ R d denotes the j-th column of the weight W ∈ R d×n and bj ∈ R n is the bias term (Deng et al., 2018). The batch size and the class number are N and n, respectively. Traditional SoftMax loss is widely used in deep face recognition (Deng et al., 2018).



Figures 2.8 Face Recognition using ArcFace (illustration)

Source: (Deng et al., 2018)

2.2.9. MTCNN (Multi-task Cascaded Convolutional Networks)

Multi-task Cascaded Convolutional Networks (MTCNN) is a framework developed for both face detection and face alignment. The process consists of three stages of convolutional networks that can recognize faces and landmark location such as eyes, nose, and mouth. The three stages of convolutional networks are :

1. Stage 1 : Stage 1 is fully convolutional Network (FCN), where FCN does not use a dense layer as part of architecture, and the FCN method called Proposal Network (P-Net), to obtain the candidate of facial windows and their bounding box, The final output of this stage is all candidate windows after refinement to downsize the volume of candidates.(K. Zhang et al., 2016).
2. Stage 2: All candidates from P-Net are fed to another CNN called Refine network (R-net), The R-Net further reduces the number of candidates, performs calibration with bounding box regression and employs non-maximum suppression (NMS) to merge overlapping candidates(K. Zhang et al., 2016).
3. Stage 3: Make output of three things: Face/Non Face classification, bounding box regression, and Facial landmark localization (Gradilla, 2020).

2.2.10. Google Cloud Platform (GCP)

Google Cloud Platform is a public cloud vendor which provides many services such as IaaS (Infrastructure as a Service), SaaS (Software as a Service), and PaaS (Platform as a Service) to be used by customers in developing applications or IT infrastructure of a company.

With Google Cloud Platform we can develop an application that we make to be serverless, can be used anywhere, and also safe. Data will be stored in Google data centers around the world for free or with pay as you go payments.

For now, the Google cloud platform has provided many features or services that users can use to develop their applications, such as Virtual Machines, Bucket (storage), App Engine (for application hosting), Databases, Networking (ex: load balancing), Big Data, and machine learning (for details of product, see this link : https://cloud.google.com/products.