Computer Vision: Face Recognition and OCR

Zara Thomas (acwg618) - City, University of London

I. Overview of Project

THIS project focuses on using state of the art techniques in Computer Vision to create a system that is primarily able to detect and identify individuals using a database of known face images. This system is also able to detect and identify unique number labels held by each individual.

A. Face Recognition System

The face recognition process begins with images being input into the system. A face detector using the Viola-Jones algorithm is then used to detect faces in the image. These faces are then cropped out and saved. The system then loops through all the faces extracted and applies either HOG or Bag of Features (feature extraction techniques) to the images. These features are then sent to the face recognition classifier where the person is then classified using either SVMs, KNN or AlexNet.

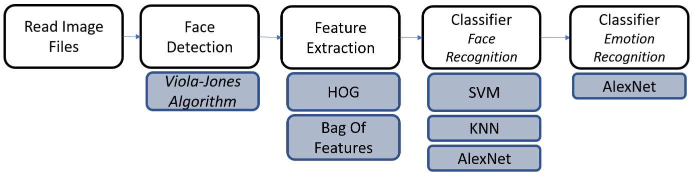

Fig. 1: Block Diagram of Face Recognition Process

B. Text Recognition System

The text recognition process begins with either images or videos being input into the system. If videos are used, frames are extracted from the videos and are then used as inputs to the system. Various image segmentation techniques are applied to the images dependent on whether individuals are holding clipboards or not. This is explained further in section IV. The aim is to extract the piece of paper with a number written on it from the image. These features are then sent to the text recognition classifier where the number displayed is then classified using AlexNet.

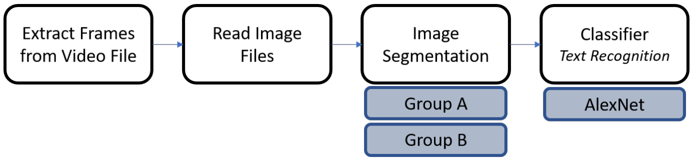

Fig. 2: Block Diagram of Text Recognition Process

MATLAB 2018a is used to develop the system because it contains a variety of toolboxes that facilitate the implementation of many of the tools and techniques used in this project. These toolboxes include the Computer Vision System, Image Processing, Statistics and Machine Learning and Neural Networks toolboxes.

C. Function Calls:

Feature Types: "HOG" or "BAG"
Classifiers: "SVM" or "KNN" or "AlexNet"

II. Face Detection

Prior to carrying out any of the required tasks, it was imperative to correctly input the images in the correct orientation i.e. with each person upright.This is vital because the system is only able to use the Optical Character Recognition function(OCR) using upright images. Further details on this is explained in section II.
The Viola-Jones algorithm is used to detect the upper bodies of individuals in order to detect faces. This was more successful than the default 'frontal face' method as people were likely to turn their heads away from the camera especially in the group photos. Detecting upper bodies was a way to solve this problem however, this method was not as successful in images where the camera is not directly facing the individual.

1) Method: Faces are detected in both individual photos and group photos. These faces are then extracted from images using the x and y co-ordinates from where the face starts and the length and width of the face. In group photos the system runs a for loop through all the detected faces and extracts the image into a local directory.

One issue that arises is that the size of faces extracted differs significantly between faces in group images(due to distance from camera) and between individual images and group images. This poses a problem when resizing as there is a loss of information when smaller images extracted from the group set must be resized to the standard image input for classifiers(set as 227x227x3).

III. Face Recognition

A. Support Vector Machines (SVM)

The first classifier experimented with was Support Vector Machines. SVMs are a form of supervised classification algorithms. Given a set of training examples, SVMs construct a hyper-plane that maximally separates the positive and negative instances in the training set. Given there are 54 classes in this dataset, multiclass SVMs are used whereby classes are separated from other classes individually. The 54 classes included one class for unlabelled('unknown') individuals detected in group photos. In object recognition, a robust feature set must be extracted from images in order to facilitate training the classifier. An ideal feature detector should be insensitive to changes in the image e.g. lighting, scale.

1) Histogram of Oriented Gradient (HOG): Histogram of Oriented Gradient descriptors have shown to be one of the best feature sets for human detection when combined with SVMs [10]. Gaussian kernels in particular have shown impressive results (95% accuracies). This method is based on using the distribution(histograms) of directions of oriented gradients as features within in an image. Gradients (i.e. derivatives of x and y) can extract information about corners and edges within an image. The change in the magnitude of gradients is larger around edges and corners than in other flat regions of an image. An advantage of using HOG features is that the capture of structure locally within an image is invariant to geometric transformations(e.g. translation and rotation of images) but are also able to adapt to changes of lighting conditions [10].

2) Method: HOG features are implemented by dividing the image into 'cells' where each cell represents a local 1-D histogram of gradient directions of the pixels within the cell. These cells are the extracted features from the image, fully representing the image.

In this experiment, it was found that a cell size of 10 x 10 pixels was optimum in encoding a lot of shape information whilst keeping the dimensionality of the feature vector fairly low. Limiting the number of dimensions in the model helps in reducing the time required to train the model.

Both gaussian and linear SVM models are evaluated to see whether it makes a difference, which kernel function is used. The dataset used is partitioned into the training set and test set (at 70% and 30% respectively).

3) Bag of Features: Bag of features corresponds to a histogram of a number of occurrences of particular image patterns(features) in a given image [4]. The method is as follows:

1) Extract features based on a feature extraction method such as HOG from each image.
2) Use the k-means algorithm to cluster the features or visual words into k mutually exclusive clusters. The resulting clusters are separated by similar characteristics[5].
3) Use the approximate nearest neighbour algorithm to construct a feature histogram for each image. Histogram bins are constructed by counting the number of features in proximity to the cluster centres (discrete classes).
4) The histogram of visual words is then used as the positive and negative samples to train the classifier in a multiclass classifier such e.g. SVM [5].

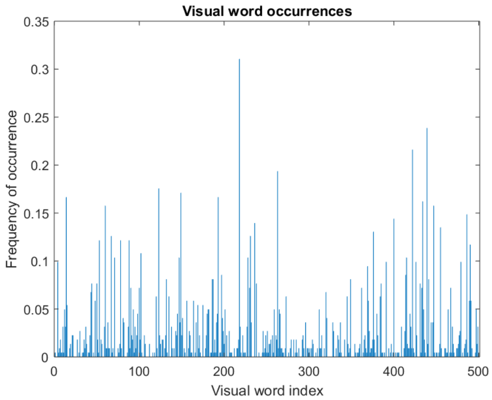

Fig. 3: Visual Word Occurences for an image

B. K-Nearest Neighbours (KNN)

KNN models when combined with HOG feature extraction have shown to perform worse than gaussian SVM models with HOG features [7].
In K-Nearest Neighbours, each object in space is represented by a position vector in a multidimensional feature space. This method is based on the nearest distance of neighbour classes. The KNN method selects the k-nearest classes depending on distance. This means that if k = 5, the algorithm will give the majority vote to its 5 nearest neighbours. In this experiment, Euclidean distance is used to measure the difference between the test point and the sample space.

C. Convolutional Neural Network (CNN)

CNNs have been shown to work well as feature extractors when using images as input without any pre-processing carried out. A CNN computes a convolved image (using a convolutional kernel), and reduces the number of parameters involved in comparison to say multilayer layer perceptrons. CNNs have been considered state-of-the-art since 2012, when AlexNet was used to win the ImageNet challenge [1].

In this experiment, transfer learning was used on a pre-trained AlexNex convolutional network to perform classification on the class dataset. Fine tuning such a network has advantages including being faster and easier to train than a network which is created from scratch as well as aiding in training datasets with a small number of samples.

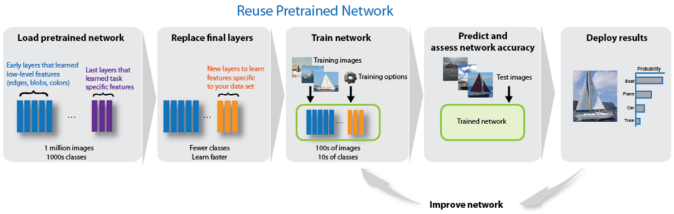

Fig. 4: Transfer learning process with CNNs [11]

A combination of strategies are used here. Firstly, the last fully connected layer are removed from the convolutional net. This layer outputs the various class scores for the data which it was trained on i.e. ImageNet. Secondly,the weights in the last layers of the CNN are fine tuned rather than the entire network. This is because earlier features of a CNN contain more generic features e.g. edge detectors that are useful in many tasks but later layers are progressively more specific to the original dataset [9].

1) Method: All layers from the AlexNet are extracted except for the last three layers which consist of a fully connected layer, a softmax layer and a classification output layer.
Data augmentation is then used to pre-process the images before being used as a input to the network to help prevent

the network from overfitting. Operations performed include randomly flipping images along the vertical axis and translating images up to 20 pixels horizontally and vertically. The images are also resized to 227x227x3 as this is a requirement for AlexNet.

D. Emotion Recognition

This functionality enables the system to detect the emotional state of a person given an image or video file. Emotional states are as follows:

Label	Emotional State
0	happy
1	sad
2	surprised
3	angry

TABLE I: Emotions and their corresponding labels

Transfer learning was also used in emotion recognition in a similar fashion to the Alexnet classifier used in Face Recognition. As the images are not labelled with which emotion is expressed by the individual, it was necessary to organize the images into four folders with the emotional states used to name each folder. The folder names could then be used as labels or targets for the classifier. It was quite challenging labelling the images as emotions expressed by individuals were at times ambiguous. Hence, only emotions that were clearly expressed were included in the dataset used to train the CNN.

IV. Image Segmentation

In the following section, the methods and techniques used in image segmentation prior to OCR are discussed. As mentioned above, image segmentation is essential in obtaining good OCR results. However, it was realised that image segmentation predominantly involved a form of trial and error method to obtain a system that would be able to perform correctly for the majority of images. The images within the dataset can be roughly divided into two groups. The first group are images where the person is holding a black clipboard and the second is where the person isn't holding a black clipboard. Through the techniques experimented with, it was learnt that these two groups would have to be treated differently in terms of tools used to achieve successful segmentation.

A. Black Clipboard Group (Group A)

This group of images were easier to deal with as the black clipboard in most cases separates the paper in the foreground from the background. These images were first converted into binary images by filtering out pixels at approximated 200 for blue,red and green channels in the RGB space. This is to perform a colour segmentation of the image to find paper-coloured pixels. The resulting binary image has 1 for one paper-coloured pixels and 0 for non-paper coloured pixels e.g. the black clipboard.

The regionprops function is then used to obtain the 'Solidity' property of a region and filter images that have a solidity greater than 0.5. This function eliminates minuscule blobs in the image. The fewer blobs there are in the images, the easier it is to extract the region of interest. These steps were sufficient for successful image extraction in Group A images. The parameters used in the method above were derived from a trial and error/process of elimination technique. Parameters were altered until the majority of images were segmented successfully.

B. Group Without Clipboard (Group B)

As these images did not have the clipboard to segregate the paper from the background,edge detection and extraction was essential. Without it, the paper and the background would essential become one blob as there was nothing to separate the two. This made it near impossible to extract the piece of paper as there was nothing to distinguish it by. There are various ways to detect edges e.g. Sobel, Canny, Prewitt, Roberts and fuzzy logic methods. These methods find object boundaries by detecting discontinuities in brightness in images. After experimenting with Sobel, Canny and Prewitt methods, the Canny method performed the best in creating an image with pronounced continuous edges around the boundaries of regions.

Initially, Otsu's method was implemented in the system as it is able to minimize the weighted within-class variance and maximise the between-class variance [6]. The advantage of this is that the surrounding pixels around edges are highlighted. Therefore, when a binary image is produced, the edges around the piece of paper are conserved. However, this method was not as successful as the Canny method in preserving edges. Prior to this, the images were resize to 1333 x 1000 pixel images. Downsizing images to made it easier to bind fragile edges as less pixels were required to connect edges after performing The images were then dilated using a 6x6 matrix of ones so that the edges were reinforced creating bolder edges around objects. This was a critical step as segmentation techniques which follows(for example, colour thresholding and morphological operations) often resulted in gaps appearing in the boundaries of objects. The colour thresholding app was then used to manipulate the hue component of images in the HSV space and filter out regions based on their Hue values. The image produced was then modified by performing a morphological opening on the image. This removed any small areas giving a cleaner image.

C. Image Extraction

Following these image segmentation steps the regionprops function is used to obtain the bounding box and area of the blobs within the image. These properties are then used to specify the criteria for the piece of paper, so that it can be extracted.

The first criterion is that the aspect ratio of the length and width of the paper must be between 1 and 1.7 for group A 0.3 and 2.2 for group B. The aspect ratio of an A4 paper is in fact 1.414 however, this discrepancy was implemented because it was noted during the study that pixel values of paper differed quite significantly between the two groups. This was thought to be because the creasing of paper was more

likely for individuals without a clipboard. These creases can be seen in some of the images in group B and this results in irregular shaped extractions of the paper. The length and widths of the paper were taken from the bounding box output of the regionprops function.
The second criterion is that the area of the blob should be between 20,000 and 250,000 pixels squared. This heuristic was found from trial and error on all the images. Although, this criteria reduced the number of relevant blobs in the image significantly and sometimes even produced the expected output (i.e. the piece of paper alone), there were often cases where further criteria were required. This was especially relevant for images in group B.Hence, identified regions were only passed onto the next stage if the above criteria were fulfilled.
Two further criterion were implemented thereafter. These criteria are as follows: a) the mean of the blob is greater than 160 and b) the length of the blob is greater than 100. Criteria a was not implemented initially because the average pixel value could only be calculated once the region was extracted from the first stage of criteria. An example of the final output can be seen in figure X.

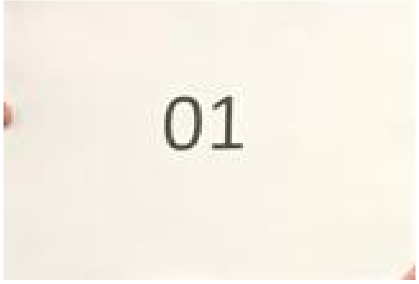

Fig. 5: Final Output after Image Segmentation and extraction

V. Optical Character Recognition (OCR)

This function accepts an image or video file of a person holding a number on a piece of paper in their hands and returns the number shown.

The OCR built-in function in Matlab is an easy way to implement text recognition however, it performs best when the text is located on a uniform background and formatted like a document [2]. In this dataset, although the numbers are printed on paper, the surrounding background is non-homogenous. Hence, additional pre-processing steps (image segmentation techniques) are required to create a uniform background before the OCR function is implemented i.e. segment out the paper with the number on it in a "frontal face" position for the OCR recognition function to work as proposed.

A. OCR Function

The OCR function was very difficult to use for the dataset. As the conditions in the images varied so much, it was difficult to output an image from segmentation which was uniform for all the digits observed. Hence, very low accuracies were observed with the OCR function.

B. Convolutional Neural Network

Due to the difficulties encountered with OCR, it was decided that AlexNet would be best suited for recognising digits in the images, given the success shown with face and emotion recognition.

VI. Results

A. Face Detection

Face detection performs relatively well in the system however, there are false positives and false negatives observed. The error rate for false positives in the group photo dataset was 0.015. This was calculated by going through the group photos and seeing what proportion of faces extracted were incorrect.

B. Face Recognition Classifier

All classifiers used in the system are evaluated and compared in this section.

Classifier	Test Error Rate
SVM & HOG	0.18
SVM & Bag of Features	0.1412
KNN & HOG	0.2729
AlexNet	0.1013

Given the success of convolutional networks in object recognition [8][1], the results for AlexNet indicate that better accuracies are indeed obtained using CNNs. The test error rate of 0.1013 is higher than the rates observed in literature. This is most likely because the dataset used here is an order of magnitude smaller thus affecting training size. The number of images within the unknown label was not as populated as the other classes. The unknown images were extracted from group photos, and videos were not used to extract more frames to increase the sample size.
Studies have shown that SVMs perform better than KNNs when combined with HOG feature extraction - achieving 96.5% and 90.73% respectively in classifying faces[7]. Again, higher test error rates are observed in this study in comparison to Nassih for the reasons mentioned above.

C. OCR Classifier

Overall the OCR Classifier works well on "ideal" images. Ideal images are those where faces are easily detected, the person is holding a clipboard, the person is not wearing white and the person is not surrounded by other objects in the background. In cases where the above criteria is not met, the OCR classifier often fails. The OCR Classifier can be broken down in to three parts a) Face Detector b) Image Segmentation and c)Text Recognition. If either of these parts fail, the OCR Classifier will not work.

1) Face Detector: The face detector fails by either producing a false negative or a false positive. In both cases, if the image is rotated the wrong way round, this will affect the image segmentation process. The criteria in this process has been defined so that the piece of paper can be extracted if landscape, however, it will fail if the paper is portrait. The criteria that is affected by this is the aspect ratio criterion and the length of the paper criterion.

2) Image Segmentation Process: The Image Segmentation process will fail if any of the criteria mentioned in section IVc is not fulfilled. One class of images which is susceptible to this in particular is class 53. The person is holding the white piece of paper whilst wearing a white/cream jumper against a white background. This is one of the extremities in the dataset. Being able to perform image segmentation on this class meant compromising the quality of segmentation in more of the other classes in the dataset. Hence, it was decided that this class would have to be exempted from the dataset.

Furthermore, if there are objects surrounding the person(in particular white objects with coloured borders or parts), the image segmentation process is likely to identify a part of the object as a region of interest. Hence, if this is extracted and passed onto the text recognition function, OCR will fail.

One of the trickiest images to deal with are those that have people holding clipboards but the paper is towards the edge of the clipboard. This results in small gaps in the boundary of the paper. These hybrid images - i.e. images with people holding clipboards but in fact behave as people without clipboards often fail in the image segmentation process. This is because the image segmentation techniques used in both cases are unable to perform well with such extremities.

3) Text Recognition: Finally, the text recognition function is also susceptible to failing. Given the relatively high test error rates of the AlexNet model, it is likely that the model with misclassify some of the images. This is because during dataset preparation some images produced had to be discarded as image segmentation wasn't good enough to extract the piece of paper.

VII. Conclusion

In conclusion, the system is able to carry out the required tasks. However, the accuracies obtained and the reliability of the system is questionable. It would be good to evaluate to what extent feature extraction methods like HOG are invariant to changes in conditions e.g. lighting and rotations. This will aid in designing a more robust system. Furthermore, although convolutional neural nets were used as classifiers in the study, the difference in accuracies between the OCR classifier and the face recognition classifier highlights the need for a large sample size before training the model.

References

[1] Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems (pp. 1097-1105).*

[2] MATLAB, Recognize Text Using Optical Character Recognition(OCR).[Online].Available: https://uk.mathworks.com/help/vision/examples/digit-classification-using-hog-features.html.[Accessed: 16-Apr-2018].

[3] N. Dalal and B. Triggs, âĂIJHistograms of oriented gradients for human detection,âĂİ in Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, 2005, vol. I, *pp.886-893*

[4] G. Csurka, C. Dance, L. Fan, J. Willamowski, and Cedric Bray, Visual categorization with bag of keypoints,Int. Work. Stat. Learn. Comput. Vis., *pp.1-22*, 2004.

[5] MATLAB, Image Classification with Bag of Visual Words. [Online]. Available: https://uk.mathworks.com/help/vision/ug/image-classification-with-bag-of-visual-words.html. [Accessed: 17-Apr-2018].

[6] J. Zhang and J. Hu, Image Segmentation Based on 2D Otsu Method with Histogram Analysis, in 2008 International Conference on Computer Science and Software Engineering, 2008,*pp. 105âĂŞ108.*

[7] B. Nassih, N. Hmina, and A. Amine, Face Classification under Different Kernel Function Compared to KNN Classifier, in Proceedings - Computer Graphics, Imaging and Visualization: New Techniques and Trends, CGiV 2016, 2016, *pp. 232âĂŞ236.*

[8] LeCun, Y., Cortes C., Burges, C., *The MNIST Database of Handwritten Digits [Online].* Available: http://yann.lecun.com/exdb/mnist [Accessed March 15, 2018.]

[9] A. Karpathy, Convolutional Neural Networks for Visual Recognition,Stanford CS class CS231n notes, 2017. [Online]. Available: http://cs231n.github.io/.

[10] N. Dalal and B. Triggs, Histograms of oriented gradients for human detection, in Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, 2005, vol. I, *pp. 886âĂŞ893.*

[11] , Transfer Learning Using AlexNet, 2018.*[Online]. Available: https://uk.mathworks.com/help/nnet/examples/transfer-learning-using-alexnet.html.* [Accessed: April 16 2018].