AMERICAN INTERNATIONAL UNIVERSITY- BANGLADESH

Unsupervised Computer Vision

by

Mahboob Annoor, Das Prosenjit, Billah Md. Mustain and
Khan Sakil Ahamed

A thesis submitted in partial fulfillment for the
degree of Bachelor of Science

in the
Computer Science and Engineering
Science Information and Technology

March 2016

Declaration of Authorship

We,Mahboob Annoor, Das Prosenjit,Billah Md. Mustain,Khan Sakil Ahamed, declare that this thesis titled, 'UNSUPERVISED COMPUTER VISION' and the work presented in it are our own. We confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: _____

Date: _____

"Do not go where the path may lead, go instead where there is no path and leave a trail."

Ralph Waldo Emerson

AMERICAN INTERNATIONAL UNIVERSITY- BANGLADESH

Abstract

Computer Science and Engineering
Science Information and Technology

Graduate Thesis Paper

by Mahboob Annoor, Das Prosenjit, Billah Md. Mustain and Khan Sakil Ahamed

In the field of computer science, machine learning emerges as new technology of Artificial Intelligence (AI). In this research, we plan to extract human sentiment from large scale Internet images such as Pinterest, Flickr and Instagram with greater accuracy. There are plenty of studies related to sentiment analysis, but we have implemented sentiment extraction technique with deep learning which is comparatively new in compare with state-of-the-art techniques. Deep learning has gained tremendous attention from machine learning and neural network community. We evaluate our approach with experimental dataset and compare with other baseline methods and we find that our technique surpass other approaches.

Acknowledgements

I offer my sincerest gratitude to my Honorable Faculty Md. Saddam Hossain, to my dearest group members .....

Contents

Abbreviations

For/Dedicated to/To my...

Chapter 1

Introduction

Sentiment analysis (also known as opinion mining) refers to the use of natural language processing, text analysis and computational linguistics to identify and extract subjective information in source materials. Sentiment analysis is widely applied to reviews and social media for a variety of applications, ranging from marketing to customer service. Existing approaches to sentiment analysis can be grouped into three main categories: knowledge-based techniques, statistical methods, and hybrid approaches [1]. Knowledge-based techniques classify text by affect categories based on the presence of unambiguous affect words such as happy, sad, afraid, and bored [7]. Some knowledge bases not only list obvious affect words, but also assign arbitrary words a probable affinity to particular emotions. Statistical methods leverage on elements from machine learning such as latent semantic analysis, support vector machines, "bag of words" and Semantic Orientation Pointwise Mutual Information (See Peter Turney's work in this area). More sophisticated methods try to detect the holder of a sentiment (i.e., the person who maintains that affective state) and the target (i.e., the entity about which the affect is felt). To mine the opinion in context and get the feature which has been opinionated, the grammatical relationships of words are used. Grammatical dependency relations are obtained by deep parsing of the text. Hybrid approaches leverage on both machine learning and elements from knowledge representation such as ontologies and semantic networks in order to detect semantics that are expressed in a subtle manner, e.g., through the analysis of concepts that do not explicitly convey relevant information, but which are implicitly linked to other concepts that do so.

Sentiment analysis has been conducted with different techniques such as lexicon based approaches, computer vision based approach, supervise based approaches. However, there are very few studies with deep learning based approach because this technology is yet to explored.

In our thesis, we first collect a image dataset with 5000 photos from different popular social networks such as Flickr, Instagram, Pinterest and so on. Then we prepare the dataset in a standard dimension according to our application. Then, we pre-process our dataset by normalizing in the scale of [0.0, 1.0]. Next, we train the dataset with deep neural network. Finally, we collect the sentiment result from our developed network. We also evaluate our techniques with standard bench mark datsets and we observe that our result improves significantly.

- We collect a large scale image dataset having 5000 Internet photos.
- We apply deep learning algorithm with popular Theano library over the dataset.
- We successfully extract sentiments (i.e., joy, anger, sad, and surprise) from our novel approach.
- Finally, we evaluate our technique with a new dataset and achieve a good accuracy.

Chapter 2

Literature Review and Background

Previously we studied about several things like Neural Networks, Deep Machine Learning, Convolutional Neural Network and Support Vector Machines. Due to the lack of accuracy in the output, majority of them were discarded. Still a little description of the mentioned topics is given bellow.

2.1 Sentiment Analysis

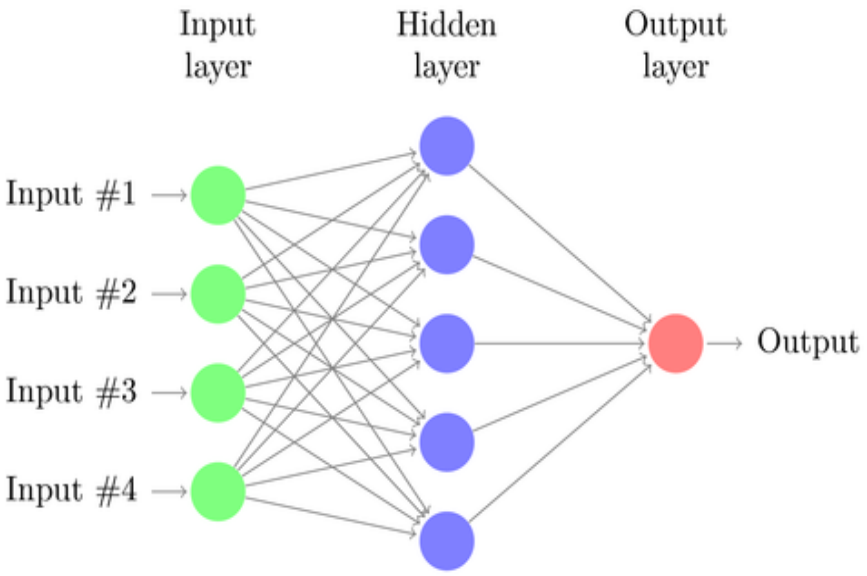
Previously we studied about several things like Neural Networks, Deep Machine Learning, Convulational Neural Network and Support Vector Machines. Due to the lack of accuracy in the output, majority of them were discarded. Still a little description of the mentioned topics is given bellow.

2.2 Neural Network

Neural networks are computational models to classify data. Application of neural networks in data classification is considerably new and it shows much promise in image classification.

In Fig 2.1, we see a basic neural network consisting of 3 layers namely input, hidden layer and output layer. In the input layer, an input is given and in the output layer, we get the desired output. The connections between the layers have weights assigned to them and each of the layers except the input layer performs a sigmoid function on

FIGURE 2.1: A simple neural network.



the calculated value for passing onto the next layer. Suppose, we have given 2 input in the input layer, Z11 and Z12 respectively. Z11 is fully connected to the next layer using weights W111, W112, W113 and W114 respectively. Z12 is also fully connected to the next layer using weights W121, W122, W123 and W124 respectively. So, the input value of the first node in the hidden layer is: Z21 = Z11* W111 + Z12* W121 Similarly, the input value of the other nodes respectively are: Z22= Z11 * W112 + Z12 * W122 Z23 = Z11* W113 + Z12* W123 Z24 = Z11* W114 + Z12* W124 Here, we can use matrix so show this efficiently. Suppose the matrix with the value of the input layer is Z1.The weight matrix for the connection between input layer and hidden layer is W1. So, the output matrix is: Z2 = Z1 * W1 Now, we have to apply a sigmoid function on the calculated values to obtain the normalized data. So, the equation for that is: Z2 = f (Z2) similarly, for the output layer: Z3 = Z2* W2, Z3 = f (Z3) Now, we have a sophisticated non-linear classifier. For measuring the values of weight matrices, W, we use a technique called back-propagation. Using supervised data set, we set the input and the subsequent output. We then progress backwards from the output and fix weights in such a way that we arrive at the input and the calculated value matches the exact value of the input. So, using back-propagation, we train the neural network[5].

2.3 Deep learning

Deep networks have been successfully applied to unsupervised feature learning for single modalities (e.g., text, images or audio). In this work, we propose a novel application of deep networks to learn features over multiple modalities. We present a series of

tasks for multimodal learning and show how to train deep networks that learn features to address these tasks. In particular, we demonstrate cross modality feature learning, where better features for one modality (e.g., video) can be learned if multiple modalities (e.g., audio and video) are present at feature learning time. Furthermore, we show how to learn a shared representation between modalities and evaluate it on a unique task, where the classifier is trained with audio-only data but tested with video-only data and vice-versa. Our models are validated on the CUAVE and AV Letters datasets on audio-visual speech classification, demonstrating best published visual speech classification on AV Letters and effective shared representation learning. Convolutional Neural Networks (CNNs) and Deep Belief Networks (DBNs) (and their respective variations) are focused on primarily because they are well established in the deep learning field and show great promise for future work [9].

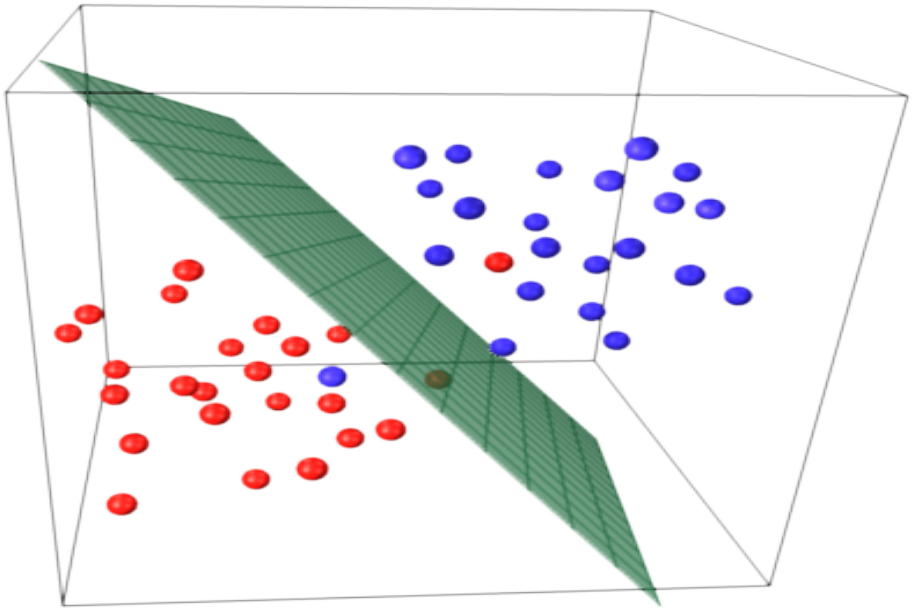
2.4 Support vector machine

Support vector machine (SVM)has been receiving increasing interest in areas ranging from its original application in pattern recognition to other applications such as regression estimation due to its remarkable generalization performance.

Support vector machines (SVMs), with their roots in Statistical Learning Theory (SLT) and optimization methods, have become powerful tools for problem solution in machine learning. Support Vector machine is a best classifier till date and the reason is Support Vector machine not only classifies the patterns it also optimizes the decision boundary, how, we will see it later but first let us refresh some of the basic concepts. It is a vector space based machine learning method where the goal is to find a decision boundary between two classes that is maximally far from any point in the training data. SVM can be classified by linear reparability and perception for linear classifiers.

SVM is a kind of large-margin classifier, it is a vector space based machine learning method where the goal is to find a decision boundary between two classes that is maximally far from any point in the training data. SVM can be classified by linear separation and perception for linear classifiers. Linearly non separable data can be solve by two approaches which is allowing a few points on the wrong side (slack variables) and map the data to a higher dimensional space. There is some learning method such as the perceptron algorithm finds just any linear separator, others like Naive Bayes search for the best linear separator according to some criterion. The SVM defines its criterion by looking for a decision surface that creates maximum distance from any data point.

FIGURE 2.2: Linearly Separable Data



Linearly Fig 2.2 non separable data can be solve by two approaches which is allowing a few points on the wrong side (slack variables) and map the data to a higher dimensional space. A linear classifier has the form f(x) = w¿x + b. In 2D the classification done by a straight-line, in 3D the discriminant is a plane and in nD it is a hyper plane.

2.4.1 Background: Neural Network

Neural networks are computational models to classify data. Application of neural networks in data classification is considerably new and it shows much promise in image classification[5]. Fig 1.1: A simple Neural Network In fig 1.1, we see a basic neural network consisting of 3 layers namely input, hidden and output layer. In the input layer, an input is given and in the output layer, we get the desired output. The connections between the layers have weights assigned to them and each of the layers except the input layer performs a sigmoid function on the calculated value for passing onto the next layer. Suppose, we give 2 inputs in the input layer, Z11 and Z12 respectively. Z11 is fully connected to the next layer using weights W111, W112, W113 and W114 respectively. Z12 is also fully connected to the next layer using weights W121, W122, W123 and W124 respectively. So, the input value of the first node in the hidden layer is: Z21' = Z11* W111 + Z12* W121 Similarly, the input value of the other nodes respectively are: Z22' = Z11 * W112 + Z12 * W122 Z23' = Z11* W113 + Z12* W123 Z24' = Z11* W114 + Z12* W124 Here, we can use matrix so show this efficiently. Suppose the matrix with the value of the input layer is Z1.The weight matrix for the connection between input layer and hidden layer is W1. So, the output matrix is: Z2' = Z1 * W1 Now, we have

to apply a sigmoid function on the calculated values to obtain the normalized data. So, the equation for that is: Z2 = f(Z2') Similarly, for the output layer: Z3' = Z2* W2 Z3 = f(Z3') Now, we have a sophisticated non-linear classifier. For measuring the values of weight matrices, W, we use a technique called back-propagation.Using supervised data set, we set the input and the subsequent output[5]. We then progress backwards from the output and fix weights in such a way that we arrive at the input and the calculated value matches the exact value of the input.So, using back-propagation, we train the neural network

2.4.2 Deep Belief Networks

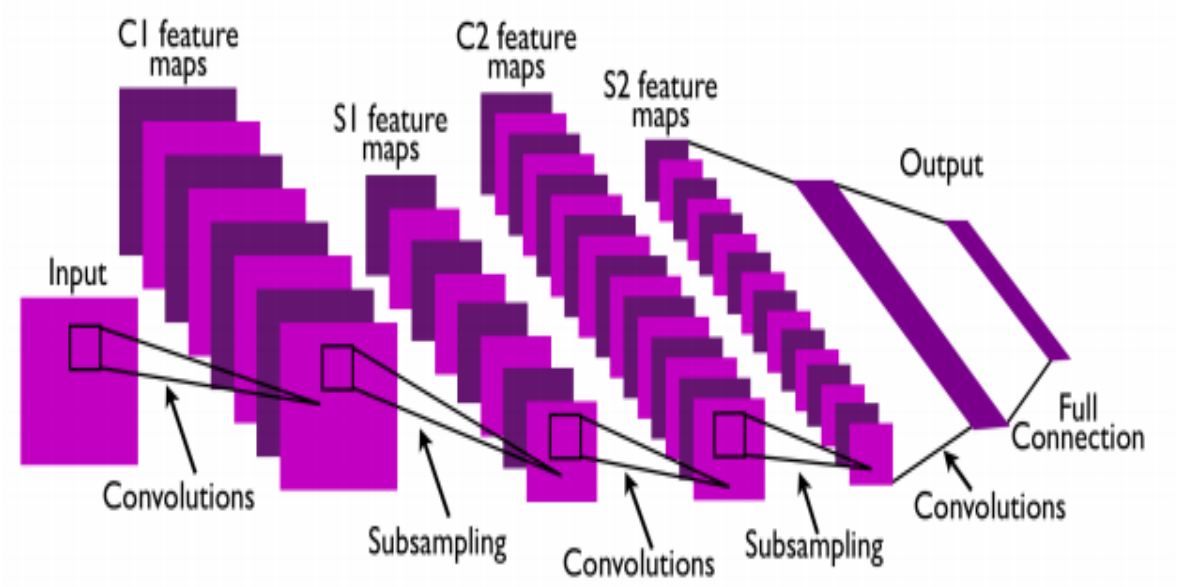
Deep networks have been successfully applied to unsupervised feature learning for single modalities (e.g., text, images or audio). In this work, we propose a novel application of deep networks to learn features over multiple modalities. We present a series of tasks for multimodal learning and show how to train deep networks that learn features to address these tasks. In particular, we demonstrate cross modality feature learning, where better features for one modality (e.g., video) can be learned if multiple modalities (e.g., audio and video) are present at feature learning time. Furthermore, we show how to learn a shared representation between modalities and evaluate it on a unique task, where the classifier is trained with audio-only data but tested with video-only data and vice-versa. Our models are validated on the CUAVE and AVLetters datasets on audio-visual speech classification, demonstrating best published visual speech classification on AVLetters and effective shared representation learning. Convolutional Neural Networks (CNNs) and Deep Belief Networks (DBNs) (and their respective variations) are focused on primarily because they are well established in the deep learning field and show great promise for future work[9].

2.4.3 Convulational Neural Network

Convolutional Networks are used to classify 2-dimensional or higher dimensional data by reducing the features to an array of small sized 2-dimensional data sets and then classifying them using neural networks.

In Fig 2.3 we see a convolutional network. The input is a image with three color channels. So, the input is a 3-dimensional array, an array of 2-dimensional data sets. These are called feature matrices. Each data set is treated as a matrix and is multiplied with lots of filter matrices to provide lots of lower dimension output matrix. This process is called convolution. These outputs are the new feature matrices. Features are extracted here. Then, the data sets are sub sampled to lower dimension. This is also called pooling.

FIGURE 2.3: A typical ConvNet architecture with two feature stages [1]



Pooling highlights dominant data and removes ambiguous datas. Pooling brings out the features. Next, a tanh function is used on all of the data to further remove ambiguous datas. The above processes comprise of one stage. These stages are repeated several times to effectively reduce the dimension a great deal.Then, the resultant 3d matrix is the input for a deep neural network. After the network is trained, input images can be classified based on desired features.

2.5 Previous Work

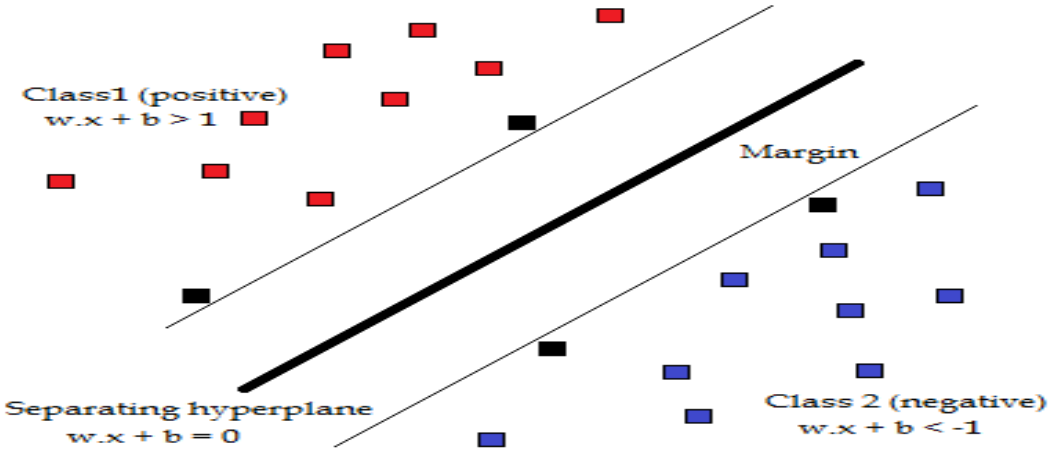
The process they have proposed and followed to solve the problem was by using a GPU based convolutional neural network. The proposed methodology describes that it will consists of a parallelized implementation of convolutional neural networks with a special emphasize on also parallelizing the detection process. The task of face detection and pose estimation and evaluating its run time performance has implemented in this paper. In detail, the topics that have focused through this paper was the extension of the face recognition (and pose estimation) system by parallelizing important parts of the computational process and after that it was implemented on a graphics card and further enhancing the system by an optimized detector. The process that followed to detect a face is it samples an input image by sliding a small window over the image with different scaling factors. While doing this sub-sampling some redundancies occurred in calculation. To gain a better run time performance these redundancies were avoided

and four different offsets were considered for sub-sampling on whether the 2x2-sub-images start with even or odd coordinates in horizontal or vertical direction respectively. Although beside the face detection its efficient implementation was another focus of the paper. According to the proposed methodology and data result set it showed a dramatic speedup compared to a CPU based implementation [6].

Support vector machines (SVMs), with their roots in Statistical Learning Theory (SLT) and optimization methods, have become powerful tools for problem solution in machine learning. SVMs reduce most machine learning problems to optimization problems and optimization lies at the heart of SVMs. SVM is a kind of large-margin classifier, it is a vector space based machine learning method where the goal is to find a decision boundary between two classes that is maximally far from any point in the training data. SVM can be classified by linear separation and perception for linear classifiers. Linearly non separable data can be solve by two approaches which is allowing a few points on the wrong side (slack variables) and map the data to a higher dimensional space. There is some learning method such as the perception algorithm finds just any linear separator, others like Naive Bayes search for the best linear separator according to some criterion. The SVM defines its criterion by looking for a decision surface that creates maximum distance from any data point. This distance from the decision surface to the closest data point determines the margin of the classifier. This method of construction necessarily means that the decision function for an SVM is fully specified by a (usually small) subset of the data which defines the position of the separator. These points are referred to as the support vectors. Here w and b is the variable, the maximize the margin M=2/——W——. Here we are assuming N training points (Xi, Yi), Yi = 1 or -1. In the figure 2.4 we are considering two planes across the classifier boundary one is plus plane and another one is minus plane. Equation of each supporting vector point will be WXi + b ¿= 1, if Yi=1(for plus plane) and WXi + b ¡= -1(for minus plane). We can unify them: Yi (WXi+b)¿= 1 [8].

In this paper the problem they have considered was face detection under pose variations. The process they have proposed and followed to solve the problem was by using a GPU based convolutional neural network. The proposed methodology describes that it will consists of a parallelized implementation of convolutional neural networks with a special emphasize on also parallelizing the detection process. The task of face detection and pose estimation and evaluating its run time performance has implemented in this paper. In detail, the topics that have focused through this paper was the extension of the face recognition (and pose estimation) system by parallelizing important parts of the computational process and after that it was implemented on a graphics card and further enhancing the system by an optimized detector.

FIGURE 2.4: Linearly Separable Data



The main focus of this paper is to present an application of back-propagation networks to hand-written digit recognition. According to this paper, previous methodologies to solve image recognition problems were lengthy and complex. A better method like large back-propagation network was applied in this paper. According to the methodology the input of the network consist of normalized images of isolated hand written digits. These inputs basically consist of black and white pixel and all the different number of ten digits will be well-separated from each other so that it could identify each digit easily. As the input of back propagation network is fixed size, a linear transformation was performed to make the characters fit in a 16 by 16 pixel image. The solution that was proposed for the problem is to scan the image with a single neuron which will have a local receptive field and store the states of this neuron in a layer called feature map. Following parallel process this feature map could be implemented as a plane of neurons whose weight vectors are constrained to be equal. The network architecture that was proposed could be trained on a low-level representation of data. Data redundancy and the constraints on the network were generates short learning time considering the size of training data set. According to the scheme this network connections and weights could be implemented by back propagation learning on commercial digital signal processing hardware[3].

A computation approach was described here for edge detection. The design was been based on the specification of detection and localization criteria in a mathematical form. Edge detectors of some kind, particularly step edge detectors, have been an essential part of many computer vision systems. The edge detection process had been served to simplify the analysis of images by drastically reducing the amount of data to be processed, while at the same time preserving useful structure information about object bound- arise. Certainly, a great deal of diversity has been noticed in the applications of edge detection but a mathematical form for those criteria which could be used to

design detectors for arbitrary edges have been developed in this paper. Another one criteria and the third criterion are added to ensure that the detector has been only one response to single edges. The criteria in numerical optimization to derive detectors for several common image features, including step edges are used here. On specializing the analysis to step edges, they found that there was a natural uncertainty principle between detection and localization performance, which are two main goals. With this principle they drive a single operator shape which was optimal at any scale. The optimal detector had a simple approximate implementation in which edges were marked at maxima in gradient magnitude of a Gaussian-smoothed image[2].

According to this paper, previous methodologies to solve image recognition problems were lengthy and complex. A better method like large back-propagation network was applied in this paper. According to the methodology the input of the network consist of normalized images of isolated hand written digits. These inputs basically consist of black and white pixel and all the different number of ten digits will be well-separated from each other so that it could identify each digit easily. As the input of back propagation network is fixed size, a linear transformation was performed to make the characters fit in a 16 by 16 pixel image. The solution that was proposed for the problem is to scan the image with a single neuron which will have a local receptive field and store the states of this neuron in a layer called feature map. Following parallel process this feature map could be implemented as a plane of neurons whose weight vectors are constrained to be equal. The network architecture that was proposed could be trained on a low-level representation of data[4].

Chapter 3

Experiment

A computation [10] approach was described here for edge detection. The design was been based on the specification of detection and localization criteria in a mathematical form.Edge detectors of some kind,particularly step edge detectors,have been an essential part of many computer vision systems.The edge detection process had been served to simplify the analysis of images by drastically reducing the amount of data to be processed, while at the same time preserving useful structure information about object boundaries.Certainly, a great deal of diversity has been noticed in the applications of edge detection but a mathematical form for those criteria which could be used to design detectors for arbitrary edges have been developed in this paper. A another one criteria and the third criterion are added to ensure that the detector has been only one response to a single edges. The criteria in numerical optimization to derive detectors for several common image features,including step edges are used here.On specializing the analysis to step edges,they found that there was a natural uncertainty principle between detection and localization performance,which are two main goals.With this principle they droved a single operator shape which was optimal at any scale.The optimal detector had a simple approximate implementation in which edges were marked at maxima in gradient magnitude of a Gaussian-smoothed image[2].

3.1 Experiment Setup

3.2 Experiment Result

3.3 Comparison With Other Approaches

The process they have proposed and followed to solve the problem was by using a GPU based convolutional neural network. The proposed methodology describes that it will consists of a parallelized implementation of convolutional neural networks with a special emphasize on also parallelizing the detection process. The task of face detection and pose estimation and evaluating its run time performance has implemented in this paper. In detail, the topics that have focused through this paper was the extension of the face recognition (and pose estimation) system by parallelizing important parts of the computational process and after that it was implemented on a graphics card and further enhancing the system by an optimized detector. The process that followed to detect a face is it samples an input image by sliding a small window over the image with different scaling factors. While doing this sub-sampling some redundancies occurred in calculation. To gain a better run time performance these redundancies were avoided and four different offsets were considered for sub-sampling on whether the 2x2-sub-images start with even or odd coordinates in horizontal or vertical direction respectively. Although beside the face detection its efficient implementation was another focus of the paper. According to the proposed methodology and data result set it showed a dramatic speedup compared to a CPU based implementation[6].

According to this paper, previous methodologies to solve image recognition problems were lengthy and complex. A better method like large back-propagation network was applied in this paper. According to the methodology the input of the network consist of normalized images of isolated hand written digits. These inputs basically consist of black and white pixel and all the different number of ten digits will be well-separated from each other so that it could identify each digit easily. As the input of back propagation network is fixed size, a linear transformation was performed to make the characters fit in a 16 by 16 pixel image. The solution that was proposed for the problem is to scan the image with a single neuron which will have a local receptive field and store the states of this neuron in a layer called feature map. Following parallel process this feature map could be implemented as a plane of neurons whose weight vectors are constrained to be equal. The network architecture that was proposed could be trained on a low-level representation of data[4].

Chapter 4

Conclusion

Bibliography

[1] E. Cambria, B. Schuller, B. Liu, H. Wang, and C. Havasi, "Knowledge-based approaches to concept-level sentiment analysis," *IEEE Intelligent Systems*, no. 2, pp. 12–14, 2013.

[2] J. Canny, "A computational approach to edge detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, no. 6, pp. 679–698, 1986.

[3] B. B. Le Cun, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in neural information processing systems*. Citeseer, 1990.

[4] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[5] Y. LeCun, K. Kavukcuoglu, C. Farabet *et al.*, "Convolutional networks and applications in vision." in *ISCAS*, 2010, pp. 253–256.

[6] F. Nasse, C. Thurau, and G. Fink, "Face detection using gpu-based convolutional neural networks," in *Computer Analysis of Images and Patterns*. Springer, 2009, pp. 83–90.

[7] A. Ortony, G. L. Clore, and A. Collins, *The cognitive structure of emotions*. Cambridge university press, 1990.

[8] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.

[9] Y. Tang, "Deep learning using linear support vector machines," *arXiv preprint arXiv:1306.0239*, 2013.

[10] V. B. Weigel, *Deep Learning for a Digital Age: Technology's Untapped Potential To Enrich Higher Education*. ERIC, 2002.