RESEARCH ARTICLE OPEN ACCESS

Object Detection in Images Using a Machine Learning based Convolutional Neural Networks (CNN)

G.Krishnaveni [1], T.Satya Nagamani [2]

[1] [2] Department of Information Technology, Sir C R Reddy College of Engineering, Eluru, Andhra Pradesh - India

ABSTRACT

As object detection is associated with video examination and picture understanding, it has pulled in much inquire about consideration as of late. Customary article identification techniques are based on carefully assembled highlights and shallow trainable structures. In this paper we proposed an enhanced CNN for object detection. The convolutional neural network for image classification and object localization had a huge impact on the computer vision community. This was not only due to the big improvement in classification performance. It also soon became clear that the convolutional layers of the network learned image features that are applicable for a wide range of vision related tasks like scene recognition and domain adaptation. ***Keywords***: CNN, object, image, detection

I. INTRODUCTION

The capacity to distinguish the objects present in a picture or scene is one of the most essential prerequisites with regards to collaborating with one's condition. While it appears to be totally easy with people and in certainty most creatures, attempting to instruct PCs to see - and furthermore understand" what they are seeing - has demonstrated amazingly troublesome. The way to understanding visual scenes are three firmly related sub-issues. The least demanding one will be called characterization in the accompanying. For arrangement the one prevailing article in a given picture ought to be resolved and marked. The following all the more requesting undertaking is object localization: notwithstanding naming the prevailing item, it additionally should be restricted in the picture, normally by deciding a jumping box around the picture area that is involved by the article. The trouble of this errand again increments if one as well as all objects in a picture should be named and various objects of a similar class can show up in one picture. This undertaking is called object discovery. Extracting higher level semantic information from images is one of the oldest and most commonly known computer vision tasks. In this thesis the problem of extracting the set of object categories present in a given image is studied. Finding an ultimate method that solves this problem has become a desired goal, mainly because of the huge amount of image data available, due to the increased popularity of various hand-held image devices. Searching in these large databases for an object of given category, by inspecting the visual cues of individual images is as an extremely challenging task in the field of computer vision. Every year the best computer vision labs submit their image classification and object detection pipelines to numerous contests that compare their system's performance (namely ImageNet Large Scale Visual Recognition Challenge (2009), Pascal Visual Objects Classes challenge (2008), Caltech- 101 (2008), etc.). In this challenging environment, even a slightest improvement of a state of the art image classification system is regarded as an interesting accomplishment. In this paper we proposed an enhanced CNN for object detection. The convolutional neural network for image classification and object localization had a huge impact on the computer vision community. This was not only due to the big improvement in classification performance. It also soon became clear that the convolutional layers of the network learned image features that are applicable for a wide range of vision related tasks like scene recognition and domain adaptation (2010). The rest of the paper is organized as follows section-2 gives the overview of state of the art done on object detection, section-3 presents the proposed mechanism, section-4 gives the performance results and section-5 concludes the paper .

A. Literature Review

The main objective of this thesis was to localize and recognize food objects in image. Several researchers have been working on this for a long time. Chen et al. (2009) prepared the Pittsburgh Fast-Food Dataset (PFFD). They selected 101 food items from different chain restaurants. Each food item was bought on three different dates. They created two baselines for the recognition task. One was based on histograms of colors and another on bag of SIFT features. Both features were later was used with SVM classifier. Shroff et al. (2008) proposed, DiaWear, a mobile or wearable camera-based semi-automatic food recognizer. They also used a reference object on the side of food item for size- reference. They worked with only 4 categories of

food, namely, Hamburger, Fries, Chicken nuggets and Apple pie. They preprocessed the image to remove the background noises and then propagated the preprocessed image through an ANN. Separately, they had experimentally achieved a threshold value for output of the network. If the output of the network was above the threshold, they accepted the result as a valid food item. They calculated contextual weights from each food and then used the Law of Total Probability for calculating the probabilities of each class. A lookup table was used to display the calorie range of highly probable food. Finally, they reported an improvement of 6.97%, 12.82%, 12.19% and 2.17% in Hamburger, Fries, Chicken Nuggets and Apple Pie, respectively on contextual setting compared to non-contextual setting in PFFD baselines. Yang et al. (2010) attempted to utilize the special relations of ingredients of food. They worked with sixty-one categories of food. Their approach to recognition was to assign the pixels to one of these categories probabilistically using the STF (2008). They extracted the pairwise statistic of local features (distance, orientation, mid-point category, between-pair categories) to create a multi-dimensional histogram. These histograms were used to classify the images with a multiclass SVM model. The researchers used the Pittsburgh Fast-food Image Dataset (2009). They used bag of SIFT and color histogram as their baselines. They reported an accuracy of upto 28%. Keiji Yanai and his research group at the University of Electro- Communications, Tokyo, have also been working on food- image recognition and automatic calorie-estimation. They have a series of academic papers on food-image recognition which are discussed as follows. Joutouet al. (2009) studied recognition of food-images with multiple kernel learning (MKL) (2006). They used multiple kernels to combine image features like bag-of- features, colour histogram, Gabor texture, SIFT-features, etc. They trained MKL-SVM models on the extracted features from the training images. They extracted features from the test images and used the trained MKLSVM models to classify the food image. The researchers achieved 61.34% classification accuracy on 50categories of food, which they claim was good enough results to be used in mobile applications. Hoashiet al. (2010) integrated seventeen kinds of different image features and used the same techniques as Joutuet al.(2009) to classify 85 categories of food items. They report 62.52% of accuracy which is an improvement, while the number of categories is much higher. Matsuda et al. (2012) attempted to take into account multiple food items in a single picture. Their approach is a manual effort of multiple stages. First they use different detectors for feature extraction: Deformable Partbased Model (DPM) (2006) is a sliding window based image pyramid which uses linear SVM for detecting object. Circle detector tries to detect circular objects in the image, mainly plates, bowls, etc. Region Segmentation is performed using JSEG algorithm (2009) which takes the number of regions needed as a parameters and returns the regions. They combine all the candidate regions returned by the aforementioned methods. They apply the work done by Joutouet al. (2012) on the combined image. They have reported the accuracy of 55.8% and 68.9% on multiple-item food images and single-item food images respectively. Kawano et al. (2013) developed a real-time mobile food recognition system using smart phone. Users are asked to take picture of food and draw a bounding box over the food item they want recognized. The image on the bounding box is further segmented using GrubCut. The image-feature-color moment, color histogram, color- auto correlogram, HoG, PHoG, Bag-of-SURF, Gabor texture feature are extracted in the next phase and the features are then used to train a linear SVM and a fast _2 kernel. The same procedure is applied first to extract features from the test images. Then these features are used to classify one of the fifty categories of images using trained linear SVM and fast _2 kernel. The authors report 81.55% of classification rate on top 5 food items if the ground truth bounding-box is supplied by the user.

II. PROBLEM DESCRIPTION

In this section, nomenclatures, the problem, and the assumptions required to model the problem are introduced before the mathematical formulation. Convolutional Neural Networks The basic structure conforming this hierarchical structure is known as a neuron. This name comes from its similarity to the biological neurons from the brain, since it tries to replicate its functionality. Figure-1 shows the structural and functional resemblances. Both of them receive several inputs which are weighted and then computed in order to produce an output.

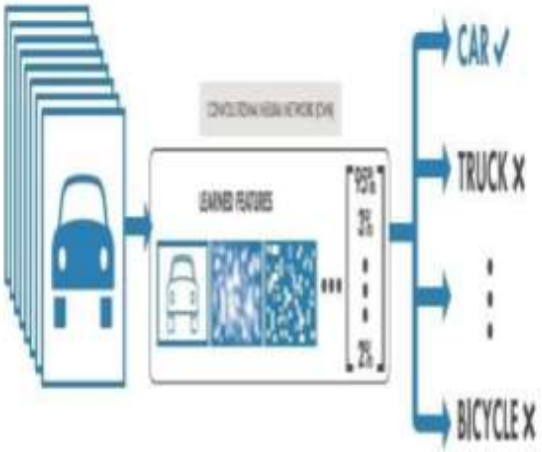

Fig. 1 CNN for object detection

Each of the input is multiplied by some parameters, called weights, and introduced to the body of the neuron. Commonly, after the computation is done a non-linearity is applied, that settles a threshold for when the neuron provides an output or not. This is why the non-linearity is also called activation function. When several of this neurons are clustered together they constitute a layer, that once is provided with the input data, computes its activations and generates an output that

most probably will be the input for another layer. When several layers are concatenated they form a neural network and those layers that are situated between other layers are called hidden units or hidden layers. In figure-1 can be seen a two layers.neural network on the left and a three layers NN on the right. In the following sections different layers are explained in order to build a better understanding of the structure of a neural network.

A.Fully Connected Layers

In the layers explained before, each of the neurons receive all the values from the previous layers or from the input data. This can be appreciated in figure-1. This structure requires a high number of connections, what means a large number of parameters and consequently a large amount of memory is required. These layers receive the name of fully connected layers and are mainly used to generate the features that deal with the class probabilities.

B. Convolutional Layers

These layers are the ones that give the name to the convolutional network structures. Its main characteristic is that they group together several values of the input, forming matrices. The weights applied to the input values receive the name of filters and they are different for each neuron. This strategy provides to the network spatial coherence and consequently makes this layer really efficient in the computer vision field. Moreover, avoiding the connection of all the values of an input to each of the neurons requires far less parameters. The inputs of a neuron are the values of a two dimensional matrix in the image. This will compute an output and move to another region of the image, covering this way all the image and generating an output. This output is called feature map and each pixel represents how the weights of that neuron (a filter) reacts to a particular region of the image.

C. Max-pooling Layers

The purpose of using pooling layers along the network is to gradually reduce the spatial sizeof the representations of the images reducing at the same time computation for the following layers. To do that a window of pixels from the representations is selected and an operation is applied. In the case of the max-pooling layers, only the highest value from the pixels of the selected window is passed to the output of the layer. Then this operation is done along all the representation. Implementation In this section how the mean squared error has been optimized in finding the objects is explained through algorithmic procedure.

D. Batch Normalization Layers

As it will be explained in future sections, during training the parameters of the layers change until they achieve an optimal representation of the images. Consequently, the distribution of the layers' inputs also change. Thus, the layers have to constantly adapt to the new distributions. In [15] they introduce the term intern covariance shift to refer to the phenomena of the changing distributions, that force to use lower learning rates, what slows down the training. The main objective of the batch normalization layers is to reduce the internal covariance shift and this way improves the training. This is done by whitening the input of each layer; i.e. forcing the mean of the images to be close to zero and its variance close to one. In the aforementioned study [15] they accelerate the training of CNN adding batch normalization layers, since it allows several changes in the network such as: increasing the learning rate and accelerating its decay, remove dropout, reduce the L2 weight regularization, and response normalization among others.

E. Dropout Layer

To avoid overfitting the network when using small datasets, in [13] they introduce the technique called drop out, that gives name to the layer. It randomly omits some of the activations of the previous layer, this prevents from having features that depend one to each other. Training a machine learning algorithm can be seen as approximating two functions y(x) and yˆ(x), where the algorithm tries to find the closest distance from y(x) to yˆ(x) in a given metric. The basic principles of training can be illustrated with a linear regression: eq-1

$$\hat{y} = w^T x$$

Here, w is a vector of parameters that the algorithm can optimize, which in a machine learning context, are called weights. They determine how features xi correlate with the output yˆ; finding the closest “distance” between yˆ and y is called predicting y from x. There are many possible ways for an algorithm to optimize the parameters. In the provided example, a possible learning method can be to minimize the mean squared error (MSE) from equation 2 on the training set eq-2

$$\mathbf{MSE} = \frac{1}{n}\sum_i (\hat{y} - y)_i^2$$

Here, n is the number of events x with features i. yˆ is called the prediction of the model on the training set. The MSE is minimized by solving the gradient with respect to weights w for 0: eq-3

$$\nabla_w \mathbf{MSE} = 0$$

To validate the training process, the MSE is also calculated for an independent validation set xval with nval events, that the algorithm does not use for training: eq-4

$$\mathbf{MSE}_{val} = \frac{1}{n_{val}}\sum_i (\hat{y}_{val} - y_{val})_i^2$$

Here, yval is the set of correct output values and yˆval the algorithm prediction for yval based on xval. The algorithm iterates the training and validation process until the error is sufficiently small, a criterion specific to the task. In general, the user has to define a model that describes the output y in terms of input x, like the linear regression above, and a learning method.

Algorithm for training CNN for detecting objects:

{

Input: data set
Output: classified objects
Step-1: split the input data into training and testing data
Step-2: take first batch of input and give to CNN trainee eq-1
Step-3: extract the features from the each batch using CNN layers
Step-4: update the weight value of each neuron
Step-5: identify the MSE using eq-2
Step-6: validate the training using eq-4
Step-7: repeat step-1 to step-6
Step-8: end
}

The algorithm explains the object detection using CNN architecture in images. It takes images and apply CNN for identifying the objects by undergone through different stages of CNN as shown in above algorithm.

III. RESULT ANALYSIS

As experimental setup we use Ubuntu 14.04 LTS OS, 4 GB RAM, 500GB HDD. for simulation of this we use ANACONDA as simulator and python-3 with tensor flow, numpy and pandas.

COCO Data set:

Microsoft COCO Detection Dataset: The Microsoft COCO object discovery dataset contains 80 article classes. We pursue [10] to utilize 80k pictures for preparing, 60k pictures for testing.

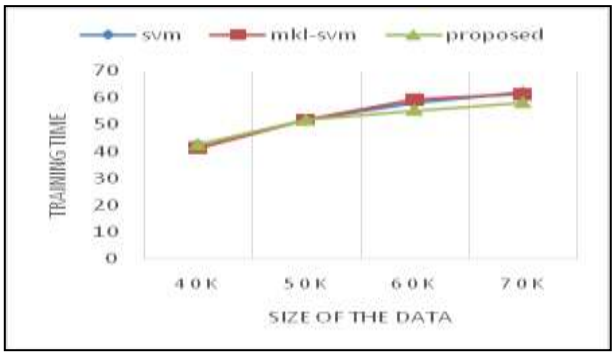

Fig.2.Training Time(Ms)

Here figure-2 shows the training time comparison of SVM and state of the art MKL-SVM and proposed method with respect to number of data samples. Here SVM takes more time initially and also time increasing with respect to data set size. And state of the art MKL- SVM takes better time with respect to data set. But proposed mechanism takes less time with respect to other two mechanisms while increasing the data set size also.

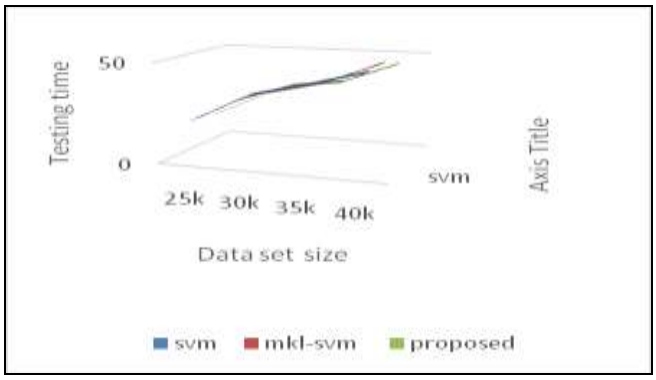

Fig. 3. Testing time (ms)

Here figure-3 shows the testing time evaluation of SVM and state of the art MKL-SVM and propose method with veneration to number of data samples. Here SVM takes more time initially and also time increasing with respect to data set size. And state of the art MKL-SVM takes better time with respect to data set. But proposed mechanism takes less time with respect to other two mechanisms while increasing the data set size also.

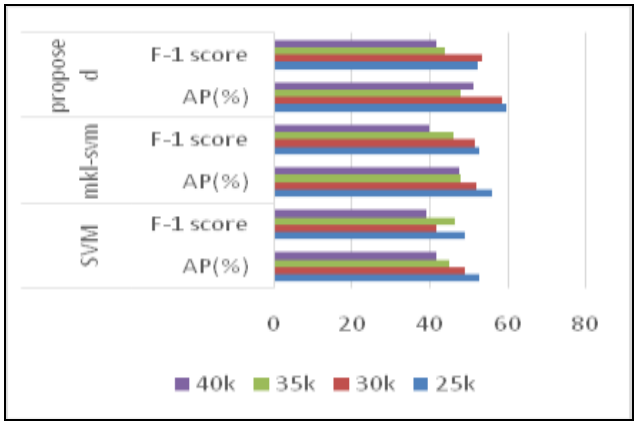

Fig.4. AP% and F1-score

Here fig-4 demonstrates the relative estimations of AP and F1-score. AP (Average exactness) is a prominent measurement in estimating the precision of article locators like SVM, MKL-SVM and proposed method. Normal exactness figures the normal accuracy esteem for review an incentive more than 0 to 1. F1 score consolidates exactness and review in respect to a particular positive class - The F1 score can be deciphered as a weighted normal of the accuracy and review. Here proposed system is outflanked than standard SVM and best in class MKL- SVM.

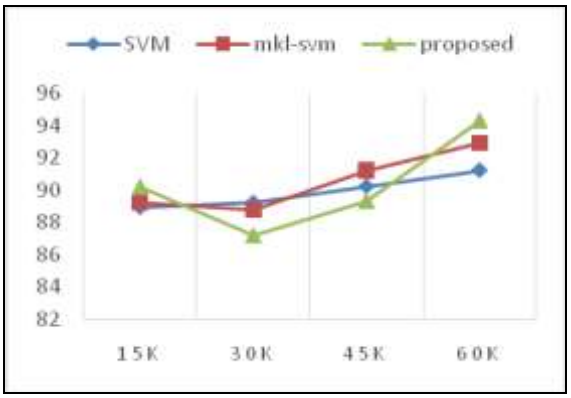

Fig. 5. Accuracy%

Here figure-5 shows the accuracy of proposed CNN and standard SVM and MKL-SVM. Accuracy refers to the exact detection of objects from an image. Here proposed mechanism outperformed the state-of the art work. Detection accuracy increase with respect to the number of images increases.

IV. CONCLUSIONS AND FUTURE WORK

In this paper we proposed a CNN based classifier for extracting and classification of objects in an image. In this paper we proposed an enhanced CNN for object detection. The convolutional neural network for image characterization and article restriction huge affected the computer vision network. This was not just because of the enormous improvement in characterization execution. It likewise before long turned out to be certain that the convolutional layers of the system learned picture includes that are material for a wide scope of vision related errands like scene acknowledgment and domain adaptation and here we used COCO image classification and also object detection challenge with positive results. Image classification using CNN produce better results comparable with current state of the art methods like SVM and its variants. It is further shown that the proposed approach improves the performance of the standard image classification architecture.

REFERENCES

[1] Abadi. M., Barham. P., Chen. J., Chen. Z., Davis. A., Dean. J., Devin. M., Ghemawat. S., Irving. G., Isard. M., et al. (2016). Tensorflow: System for large-scale machine learning, *OSDI.* 16, 265–283.

[2] Bastien.F., Lamblin. P., Pascanu. R., Bergstra. J., Goodfellow.I., Bergeron. A., Bouchard. N., Warde- Farley. D., Bengio. Y., (2012). Theano: new features and speed improvements. *arXivpreprint arXiv:1211.5590.*

[3] Bay. H., Ess. A., Tuytelaars. T., Van Gool. L., (2008).Speeded-uprobust features (surf). *Computer vision and image understanding* . 346–359.

[4] Bengio. Y., (2012).Deep learning of representations for unsupervised and transfer learning. In *Proceedings of ICML Workshop on Unsupervisedand Transfer Learning*. 17–36.

[5] Bishop. C., (1995). Neural networks for pattern recognition.Oxford universitypress,.Brosnan. T., Sun. D., (2004). Improving quality inspection of food products by computer vision—-a review, *Journal of food engineering. 61*,1, 3–16.

[6] Brynjolfsson. E., McAfee. A., (2011). The big data boom is the innovation story of our time, *The Atlantic. 21.*

[7] Chen. M., Dhingra. K., Wu. W., Yang. L., Sukthankar. R., Yang. J.,*(2009)* Pfid: Pittsburgh fast-food image dataset. In *ImageProcessing (ICIP), 16th IEEE International Conference* . 289–292.

[8] Deng. Y., Manjunath. B., (2001). Unsupervised segmentation of color texture regions in images and video, IEEE transactions on pattern analysis and machine intelligence. 23, 800–810.

[9] Felzenszwalb. P., Girshick. R. B., McAllester. D., Ramanan. D., (2010). Object detection with discriminatively trained part based models, *IEEE transactions on pattern analysis and machine intelligence. 32*, 1627–1645.

[10] Hoashi. H., Joutou. T., Yanai. K., (2010). Image recognition of 85 food categories by feature fusion, *ISM IEEE International Symposium on* IEEE. 296– 301.

[11] Joutou. T., Yanai. K., (2009). A food image recognition system with multiple kernel learning, *ICIP, 16th IEEEInternational Conference* . 285–288.

[12] Kawano. Y., Yanai. K., (2013). Real-time mobile food recognitionsystem, In *Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE Conference* . 1–7.

[13] Matsuda. Y., Hoashi. H., Yanai. K., (2012). Recognition of multiple food images by detecting candidate regions, In *Multimedia and Expo, IEEE International Conference.* 25–30.

[14] Shotton. J., Johnson. M., Cipolla. R., (2008).Semantic text on forests for image categorization and segmentation, In *Computer vision and pattern recognition,CVPR, IEEE Conference.* 1–8.

[15] Shroff. G., Smailagic. A., Siewiorek. D. P., (2008). Wearable context-aware food recognition for calorie monitoring, In *WearableComputers, ISWC , 12th IEEE International Symposium*. 119–120.

[16] Sonnenburg. S., Ratsch. G., Schafer. C., Scholkopf. B., (2006). Large scale multiple kernel learning,*Journal of Machine Learning Research.* 1531–1565.

[17] Yang. S., Chen. M., Pomerleau. D., Sukthankar. R., (2010). Food recognition using statistics of pairwise local features, In *Computer Vision and Pattern Recognition CVPR, IEEE*. 2249–2256.