

Object Recognition with Improved Features Extracted from Deep Convolution Networks

J. Siva Ramakrishna¹, Amarendra Reddy Namburi²

^{1,2}Institute of Aeronautical Engineering, Hyderabad, India

Abstract: Object recognition is a process for identifying a specific object in an image. Object recognition algorithms depends on matching, learning, or pattern recognition algorithms using appearance-based or feature-based techniques. Object recognition techniques include feature extraction and machine learning models, deep learning models such as CNN. Deep learning with convolution neural networks (CNN) has been proved to be very effective in feature extraction. CNN is comprised of one or more convolutional layers and then followed by one or more fully connected layers. In Image classifications, the work with classifiers aims at exploring the most appropriate classifiers for high level deep features. The features extracted from the image play an important role in image classification. Feature extraction is the process of retrieving the important data from the raw data. Feature extraction is finding the set of parameters that recognize the object precisely and uniquely. In feature extraction, each character is represented by a feature vector, which becomes its identity. The major goal of feature extraction is to extract a set of features, which maximize the recognition rate. In this work, the features extracted from CNN applied as input to train machine learning classifier and perform image classification. A systematic comparison between various classifiers is made for object recognition.

Keywords: object recognition; Deep learning; image classification; support vector machine; extreme learning machine;

1. Introduction

Object recognition system finds specific object in an image using object recognition techniques. Object recognition is difficult in case of defining an object with different backgrounds. In this paper, we will discuss object recognition techniques provides feature extraction, machine learning models and deep learning models such as Convolution Neural Networks (CNN). In machine learning, feature is defined as process of studying the characteristic of an image or object being observed. Unique features perform an essential function in pattern recognition and classification. Feature extraction is process of retrieving the important data from raw data. Feature extraction is locating the set of parameters that apprehend the object precisely and uniquely. The main purpose of feature extraction is to extract a set of functions, which maximize the recognition rate. A convolution neural network (CNN) is a category of deep, feed ahead artificial neural networks that has efficiently has been carried out to reading visible imaginary (Fig. 1). CNN are used to recognize images by transforming the input image via layers. CNN has visual cortex. CNN consists of collection of layers of neurons get activated. Each layer will detect the features of input image such as edge detection. The excessive stage of layers will discover greater complex capabilities in order to recognize the input image.

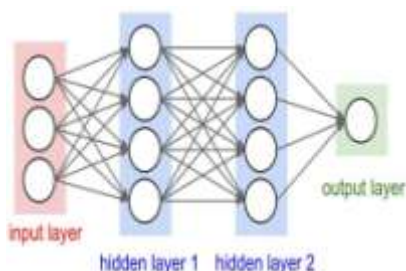


Figure 1: Convolution Neural Network

Image classification is a process to categorize all pixels in a virtual image into one among several land cover classes, or

themes. This categorised statistics may additionally then be used to produce thematic maps of the land cover present in an picture. In the terminology of machine learning classification is considered an example of supervised learning i.e. mastering where a training set of correctly identified observations is available. Classifier is a set of rules that implements class particularly in a concrete implementation.

2. Related Work

Object recognition is the hardest part of image processing. Object recognition is an important part of computer vision due to the fact it's miles intently associated with the success of many computer vision applications. Some of object recognition algorithms and systems have been proposed for a long time with a view to cope with recognition problems. A survey of different strategies inside the discipline of computer vision and object recognition are discussed. Object recognition is essential for lots of applications in the area of independent structures or business manages. Several ideas of various object recognition approaches such as feature based methods and characteristic-primarily based strategies are presented. The capabilities of present day item popularity algorithms, which belong to appearance based totally methods are discussed. SIFT (scale invariant feature transform) and SURF (speeded up robust features) both are well known algorithms of the feature-based strategy [1].

Object recognition is a critical challenge in image processing and computer vision. Affine scale invariant feature transform (ASIFT) is a method for object recognition with complete boundary detection and a vicinity merging algorithm [2]. ASIFT is a completely affine invariant algorithm which means that capabilities are invariant to 6 affine parameters specifically translation (2 parameters), zoom, rotation and two camera axis orientations. The features are very reliable and supply us strong key points that may be used for matching between different images of an object. An object in several images is skilled with one of

kind factors for finding excellent key points of it. A robust area merging algorithm is the used to recognize and detect the object with complete boundary in the different images primarily based on ASIFT key points and a similarity degree for merging areas within the image. Image segmentation is to partition an picture into significant areas with appreciate to a selected application. Previous paper presents the interaction among Image segmentation using different edge detection methods and object recognition [3]. Edge detection techniques which include Sobel, Prewitt, Roberts, Canny, and Laplacian of Gaussian (LOG) are used for segmenting the image. Expectation-Maximization (EM) algorithm, OSTU and genetic algorithms have been used to demonstrate the synergy among the segmented images and object recognition.

3. Proposed Work

Deep learning is the hottest learning technique has been widely explored in machine learning, computer vision [4]. Deep learning is a branch of machine learning which uses multiple, nonlinear processing layers to learn useful representations of features directly for data (Fig. 2). Features extracted from CNN applied as input to train machine learning classifier. A systematic comparison between various classifiers is made for object recognition. In this paper, we present a study of Nearest Neighbour (NN) [5], support Vector Machine (SVM) [6], least Square support Vector Machine (LSSVM) [7], extreme Learning Machine (ELM) [8], kernel ELM (KELM) [9] for object recognition on the deep convolutional activation features trained by CNN. Convolution Neural Networks (CNN) are composed of several inter-connected hidden layers which process and transforms inputs to outputs. They are inspired from the biological structure of the visual cortex (the part of the brain responsible for sight). Convolution layers map inputs to certain neurons in different regions (Fig. 4). Convolution layers account for non-linearity in the model. Pooling layers down sample the data to reduce the number of inputs to the next layer. The final layer is fully connected, mapping to each input into single output.

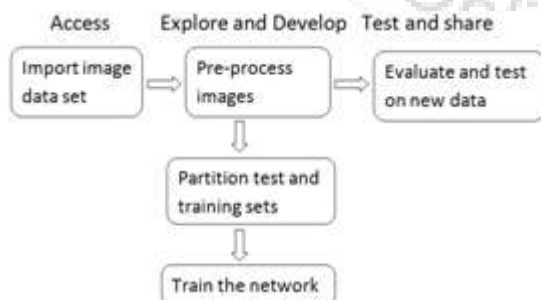


Figure 2: Deep learning Workflow

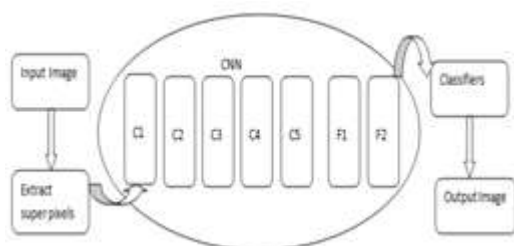


Figure 3: Block Diagram for Object Recognition

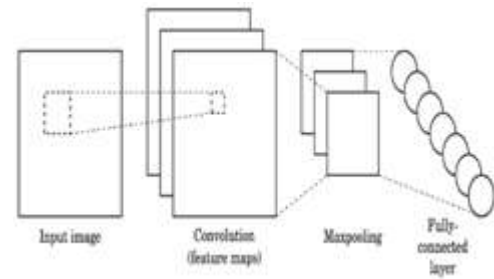


Figure 4: Convolutional layers

4. Classification

In this paper, we investigated the classification ability of deep representation on different domain data. Deep features extracted are applied to train classifiers Support Vector Machine (SVMs) and Extreme Learning Machine (ELMs). SVMs/ELMs training, and compare the classification accuracy.

a) Support Vector Machine

The principle of SVM for image classification problems is revised [10]. Given a training set of N data points $\{x_i, y_i\}_{i=1}^N$, where the label $y_i \in \{-1, 1\}$, $i = 1, \dots, N$. SVM aims to solving the risk bound minimization problem with inequality constraint.

$$\min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i \quad \text{s.t. } \xi_i \geq 0, y_i [w^T \phi(x_i) + b] \geq 1 - \xi_i \quad (1)$$

Where $\phi(\cdot)$ is a linear or nonlinear mapping function, w and b are the parameter of classifier hyper-plane. The original problem of SVM can be transformed into its dual formulation with equality constraint by using Lagrange multiplier method.

$$L(w, b, \xi_i, \alpha_i, \lambda_i) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i (2 - \sum_{i=1}^N \alpha_i (y_i [w^T \phi(x_i) + b] - 1 + \xi_i - i = 1/N \lambda \xi_i)$$

Where $\alpha_i \geq 0$ and $\lambda_i \geq 0$ are Lagrange Multipliers. The solution can be given by the saddle point of Lagrange function (2) by solving

$$\max_{\alpha_i} \min_{w, b, \xi_i} L(w, b, \xi_i, \alpha_i, \lambda_i) \quad (3)$$

By calculating the partial derivatives of Lagrange function (2) with respect to w , b and ξ_i , one can obtain

$$\begin{cases} \frac{\partial L(w, b, \xi_i, \alpha_i, \lambda_i)}{\partial w} = 0 \rightarrow w = \sum_{i=1}^N \alpha_i y_i \phi(x_i) \\ \frac{\partial L(w, b, \xi_i, \alpha_i, \lambda_i)}{\partial b} = 0 \rightarrow \sum_{i=1}^N \alpha_i y_i = 0 \\ \frac{\partial L(w, b, \xi_i, \alpha_i, \lambda_i)}{\partial \xi_i} = 0 \rightarrow 0 \leq \alpha_i \leq C \end{cases} \quad (4)$$

Then one can rewrite (3) as

$$\max \sum_{i=1}^N \alpha_i - \frac{1}{2} y_i y_j \alpha_i \alpha_j \phi(x_i)^T \phi(x_j) \quad \text{s.t. } \sum_{i=1}^N \alpha_i y_i = 0, 0 \leq \alpha_i \leq C \quad (5)$$

By solving α of the dual problem (5) with a quadratic programming, the goal of SVM is to construct the following decision function (classifier),

$$f(x) = \text{sgn}(\sum_{i=1}^M \alpha_i y_i k(x_i, x) + b) \quad (6)$$

where $k(\cdot)$ is a kernel function.

b) Least Square Support Vector Machine:

LSSVM is an improved and simplified version of SVM. LSSVM can be formulated as

$$\min \frac{1}{2} \|w\|^2 + C \cdot \frac{1}{2} \sum_{i=1}^N \xi_i^2 \quad (7)$$

$$s. t. \quad y_i [w^T \varphi(x_i) + b] = 1 - \xi_i, i = 1 \dots N$$

The details can be referred to [11]

The Lagrange function of (7) can be defined as

$$L(w, b, \xi_i, \alpha_i) = \frac{1}{2} \|w\|^2 + C \cdot \frac{1}{2} \sum_{i=1}^N \xi_i^2 - \sum_{i=1}^N \alpha_i (y_i [w^T \varphi(x_i) + b] - 1 + \xi_i) \quad (8)$$

where is the Lagrange multiplier. The optimality conditions can be obtained by computing the partial derivatives of (8) with respect to the four variables as

$$\begin{cases} \frac{\partial L(w, b, \xi_i, \alpha_i)}{\partial w} = 0 \rightarrow w = \sum_{i=1}^N \alpha_i y_i \varphi(x_i) \\ \frac{\partial L(w, b, \xi_i, \alpha_i)}{\partial b} = 0 \rightarrow \sum_{i=1}^N \alpha_i y_i = 0 \dots \dots \dots (9) \\ \frac{\partial L(w, b, \xi_i, \alpha_i)}{\partial \xi_i} = 0 \rightarrow 0 \leq \alpha_i \leq C \\ \frac{\partial L(w, b, \xi_i, \alpha_i)}{\partial \alpha_i} = 0 \rightarrow (y_i [w^T \varphi(x_i) + b] - 1 + \xi_i) \end{cases}$$

The equation group (9) can be written in linear equation as

$$\begin{pmatrix} I_0 & 0 & -Z^T \\ 0 & 0 & -Y^T \\ 0 & 0 & C I - I \xi \\ Z & Y & I \alpha \xi \end{pmatrix} \begin{pmatrix} w \\ b \\ \xi \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \dots \dots \dots (10)$$

Where $z = [\varphi(x_1) y_1 \dots \varphi(x_N) y_N]^T$

$Y = [y_1, \dots, y_N]^T$,

$\xi = [1, \dots, 1]^T, \xi = [\xi_1, \dots, \xi_N]^T$,

$\alpha = [\alpha_1, \dots, \alpha_N]^T$

The solution α and b can also be given by

$$\begin{bmatrix} 0 & -Y^T \\ Y & ZZ^T + C^{-1} I \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \dots \dots \dots (11)$$

Let $\Omega = ZZ^T$ with the mercer condition, there is

$$\Omega_{k,l} = y_k y_l \varphi(x_k)^T = y_k y_l k(x_k, x_l), k, l = 1, \dots, N \dots (12)$$

By substituting (12) into (11), the solution can be obtained by solving a linear equation instead of a quadratic programming problem in SVM.

c) Extreme Learning Machine

ELM aims to solve the output weights of a single layer feed-forward neural network (SLFN) by minimizing the squared loss of predicted errors and the norm of the output weights in both classification and regression problems. The principle of ELM for classification problems [12]. Given a dataset of N samples with label y where d is the dimension of sample and c is the number of classes. Note that if $x_i (i = 1 \dots N)$ belongs to the k -th class, the k -th position of $x_i (i = 1 \dots N)$

is set as 1, and -1 otherwise. The hidden layer output matrix H with L hidden neurons can be computed as

$$H = \begin{bmatrix} h(w_1^T x_1 + b_1) & h(w_2^T x_1 + b_2) & \dots & h(w_L^T x_1 + b_L) \\ \vdots & \vdots & \ddots & \vdots \\ h(w_1^T x_N + b_1) & h(w_2^T x_N + b_2) & \dots & h(w_L^T x_N + b_L) \end{bmatrix} \quad (13)$$

ELM can be formulated as follows

$$\min_{\beta \in R^{L \times c}} \frac{1}{2} \|\beta\|^2 + C \frac{1}{2} \sum_{i=1}^N \|\varepsilon_i\|^2 \dots \dots \dots (14)$$

$$s. t. h(x_i) \beta = t_i^T - \varepsilon_i^T, i = 1, \dots, N \leftrightarrow HB = T^t - \varepsilon^T$$

The closed form solution β of (14) can be easily solved. First, if the number N of training patterns is larger than L , the gradient equation is over-determined, and the closed form solution of (14) can be obtained as

$$\beta^* = H^* T = (H^T H + \frac{I_{L \times L}}{C})^{-1} H^T T \dots \dots \dots (15)$$

where $I_{L \times L}$ denotes the identity matrix with size of L , and is the Moore-Penrose generalized inverse of H . If the number N of training patterns is smaller than L , an under-determined least square problem would be handled. The solution of (14) can be obtained as

$$\beta^* = H^* T = H^T (HH^T + \frac{I_{N \times N}}{C})^{-1} T \dots \dots \dots (16)$$

where $I_{N \times N}$ denotes the identity matrix. Then the predicted output of a new observation z can be computed as

$$y = h(z) \beta^* = \begin{cases} h(z) \cdot (H^T H + \frac{I_{L \times L}}{C})^{-1} H^T T, & \text{if } N \geq L \\ h(z) \cdot H^T (HH^T + \frac{I_{N \times N}}{C})^{-1} T, & \text{if } N < L \end{cases} \dots (17)$$

d) Kernelized Extreme Learning machine:

The KELM can be described as follows [13]. Let $\Omega = HH^T \in \mathbb{R}^N \times N$ where $\Omega_{i,j} = h(x_i) h(x_j)^T = k(x_i, x_j)$ and $k(\cdot)$ is the kernel function. With the expression of solution β (16), the predicted output of a new observation z can be

$$\begin{aligned} y &= h(z) \beta^* \\ &= h(z) \cdot H^T (HH^T + \frac{I_{N \times N}}{C})^{-1} T \dots \dots \dots (18) \\ &= \begin{bmatrix} k(z, x_1) \\ \vdots \\ k(z, x_N) \end{bmatrix}^T (\Omega + \frac{I_{N \times N}}{C})^{-1} T \end{aligned}$$

The decision function of KELM can be expressed uniquely in (18)

5. Dataset

Examples of object images from two sources: Amazon (1st row), DSLR (2nd row). Different visual cues such as camera viewpoint, resolution, illumination, and background have been well illustrated. This paper performs a cross-domain recognition task [14]. In this, we train a SVM/ELM on the Amazon and test on DSLR, i.e. $A \rightarrow D$.



Figure 5: Images from Amazon and DLSR

6. Results

The comparison of two classifiers, the Extreme Learning Machine (ELM) and the Support Vector Machine (SVM) is considered for performance. Experiment demonstrates that the ELMs outperform SVMs in cross-domain recognition tasks. Experimental results clearly demonstrate that ELMs outperform SVM based classifiers in different settings. In particular, Kernel ELM (KELM) shows state-of-the-art recognition performance among the presented 5 popular classifiers.

Table 1: Comparison of classification accuracy

Trained Image (Amazon)	Tested Image (DLSR)	Accuracy	
		Method	A -> D
		NN	78.7± 0.59
		SVM	80.5± 0.79
		LSSVM	82.5± 0.54
		ELM	82.59±0.54
		KELM	83.9±0.44

Recognition accuracies of different cross- domain tasks by using NN, SVM, LSSVM, ELM and KELM on the deep convolutional activation features.

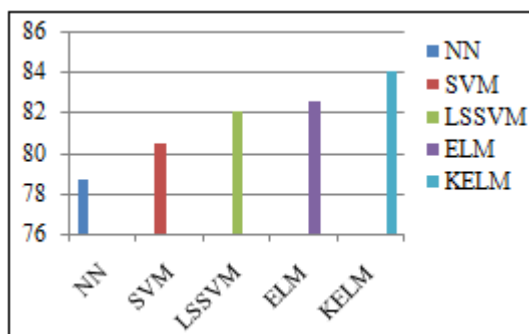


Figure 6: Classification accuracy of various learning techniques

Table 2: Comparative study of object recognition techniques

Methods	Accuracy Rate	Time Efficiency	User Rate
Deep Learning	High (85%)	High	80%
Affine scale invariant feature transform (ASIFT)	Moderate (76%)		60%
Background subtraction	Moderate	Moderate	40%
Optical Flow	Moderate	High	20%
Frame Differencing	High	Low to Moderate	30%

7. Conclusion

In this paper, All the main terminology of object detection have been addressed. These include object detection methods, feature selection and object classification. Features extracted from CNN applied as input to train machine learning classifier. A systematic comparison between various classifiers is made for object recognition. The comparison of two classifiers, the Extreme Learning Machine (ELM) and the Support Vector Machine (SVM) is considered for performance. Experiment demonstrates that the ELMs outperform SVMs in cross-domain recognition tasks.

References

- [1] Panchal, P.M., Panchal, S.R. and Shah, S.K., 2013. A comparison of SIFT and SURF. *International Journal of Innovative Research in Computer and Communication Engineering*, 1(2), pp.323-327.
- [2] Oji, R., 2012. An automatic algorithm for object recognition and detection based on ASIFT keypoints. *arXiv preprint arXiv:1211.5829*.
- [3] Ramadevi, Y., Sridevi, T., Poornima, B. and Kalyani, B., 2010. Segmentation and object recognition using edge detection techniques. *International Journal of Computer Science & Information Technology (IJCSIT)*, 2(6), pp.153-161.
- [4] CLAESSEN, L. and HANSSON, B., Deep Learning Methods and Applications.

- [5] Keller, J.M., Gray, M.R. and Givens, J.A., 1985. A fuzzy k-nearest neighbor algorithm. *IEEE transactions on systems, man, and cybernetics*, (4), pp.580-585.
- [6] Osuna, E., Freund, R. and Girosi, F., 1997, September. An improved training algorithm for support vector machines. In *Neural Networks for Signal Processing [1997] VII. Proceedings of the 1997 IEEE Workshop* (pp. 276-285). IEEE.
- [7] Suykens, J.A. and Vandewalle, J., 1999. Least squares support vector machine classifiers. *Neural processing letters*, 9(3), pp.293-300..
- [8] Huang, G.B., Zhu, Q.Y. and Siew, C.K., 2004, July. Extreme learning machine: a new learning scheme of feedforward neural networks. In *Neural Networks, 2004.Proceedings.2004 IEEE International Joint Conference on* (Vol. 2, pp. 985-990).IEEE.
- [9] Huang, G.B., Ding, X. and Zhou, H., 2010. Optimization method based extreme learning machine for classification. *Neurocomputing*, 74(1-3), pp.155-163.
- [10] V. Vapnik, "Statistical learning theory," John Wiley: New York, 1998.
- [11] J.A.K. Suykens and J. Vandewalle, "Least Squares Support Vector Machine Classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293-300, 1999.
- [12] Huang, G.B., Zhou, H., Ding, X. and Zhang, R., 2012. Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(2), pp.513-529.
- [13] Deng, W.Y., Zheng, Q.H. and Wang, Z.M., 2014. Cross-person activity recognition using reduced kernel extreme learning machine. *Neural Networks*, 53, pp.1-7.
- [14] Gopalan, R., Li, R. and Chellappa, R., 2011, November. Domain adaptation for object recognition: An unsupervised approach. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (pp. 999-1006).IEEE.