ARTICLE

BINARY SHAPE CLASSIFICATION USING CONVOLUTIONAL NEURAL NETWORKS

Habibollah Agh Atabay*

Department of Computer, Gonbad Kavous University, Gonbad Kavous, IRAN

ABSTRACT

Recognizing objects by their shapes, has been an interesting problem in image understanding, and has attracted researchers' attention. The superiority of Convolutional Neural Networks (CNNs) in various object recognition tasks have been proven in recent years. But CNNs has been rarely used in binary shape classification. In this paper, a CNN architecture is proposed specifically for the task of binary shape classification. Two already known CNN architectures along with the proposed one are trained to recognize category of binary shapes. The effect of simple types of augmentations on the final classification accuracy is presented too. The comparison of the final results with the latest best results on Animal and Mpeg7 shape datasets reflects the effectiveness of CNNs on shape recognition task even on the relatively small datasets.

KEY WORDS
Shape, Classification, Recognition, Convolutional Neural Network

Published: 16 October 2016

***Corresponding Author**
Email:
atabay@gonbad.ac.ir
Tel.: +98-89353038148

INTRODUCTION

Shape is a remarkable cue for human to understand objects in images. It can be used in image [1] and shape retrieval [2]. Shapes do not have brightness, color and texture information and are only represented by their silhouettes. In fact, shapes are stable to the variations in object color and texture and light conditions. Due to such advantages, recognizing objects by their shapes has been an interesting problem for researchers to investigate. Shape recognition is usually considered as a classification problem where a classifier is trained with a set of inputs and labels, and the trained model is tested to recognize the categories of unseen inputs. The main challenge in shape recognition is the large intra-class variations induced by deformation, articulation and occlusion. Therefore, a variety of research efforts have been made in the recent years to form an informative and discriminative shape representation.

Current approaches related to shape representation, mainly have focused on designing low-level shape descriptors which are aimed to be robust to rotation, scaling and deformation of shapes. A good shape descriptor should be invariant under rigid motion, and robust to noise and occlusion. There are two main groups of shape descriptors [3]: region-based and contour-based. The region-based method considers the information of the pixels within the shape. In contrast, the contour-based method considers the information obtained from the shape contour.

Discriminative feature learning from large datasets provides an alternative way to construct feature representations. This method has been widely used in computer vision and image processing. Bag of words (BoW) models represent images as a collection of local features. These models have shown a remarkable performance in multiple domains including object and scene recognition [4], image retrieval [1] and shape classification [5], in past decade. The major drawback of bag of words method is the lack of spatial information in the histogram representation.

While the performance of conventional, handcrafted features have plateaued in recent years, new developments in deep compositional architectures have kept performance levels rising [6]. The strong attention to deep learning is due to its capacity to create multilevel representations of data and permit for a possibly more efficient way to solve some problems. Several approaches have been presented in this area: deep auto-encoder [7], convolutional neural network [6, 8] and restricted Boltzmann machine [9] are among some of these.

CNNs have been shown to be particularly adapted to process image data, since they are inspired in the way the human visual system works [10] and are discriminatively trained via back-propagation through layers of convolutional filters and other operations such as rectification and pooling.

Although deep neural networks have attracted enthusiastic interest within various vision problems, there exists some domains that are still rarely used these powerful networks. while some recent works have used CNNs in various closely related tasks such as 3D shape recognition [11] and sketch classification [12, 13], the direct use of CNNs in binary shape classification still have not been presented. In this paper, the use of CNNs in this area is examined and is demonstrated that how CNNs can outperform current state of the art in 2D shape category recognition. In order to carry out these, various experiments on already used CNN architectures, is conducted to apply on shape datasets including Animal [14] and MPEG7 [15]. Also a new architecture is proposed and evaluated on these datasets. Steps are as follows. First, the images of dataset are padded and resized to fit into 32×32 pixels, then regular augmentation types including reflection and shift are applied. After splitting to train, validation and test sets, on each train set with specific augmentation type, two well-known CNN architectures, LeNet [16] and MNIST-CNN [17], as well as the proposed one, are trained. The results are compared with the recent achievements of conventional methods. It is observed that CNNs not only outperforms currents methods but can be used to work with small sized shape datasets with help of regular augmentations.

RELATED WORKS

In this section, the recent improvements in 2D shape classification using conventional methods and deep learning approaches were reviewed. Prior to renovation of deep neural networks and domination of these methods in object recognition tasks [6], the bag of words method has been the dominant method for summarization of large amount of local features to use in discrimination of objects and scenes. One of the applications of this method on shape classification was presented in [5], where the log-polar transform (LPT), after applying the Fourier transform modulus to enforce scale and rotation invariance, was used as a local descriptor and the bag of words framework with some novelties was used to generate feature vectors to train SVM classifier. And a method was presented to construct a low-dimensional histograms of bi-grams using the spatial co-occurrence matrix, as well as a metric to select an appropriate codebook size in the bag-of-words model and methods to estimate codebook size and maximum radius of the log-polar transform.

Contour [18] and skeleton based [19] descriptors have been among the favorite shape representations in recent years. In [18], a shape representation called Bag of Contour Fragments (BCF) was developed, inspired by classical Bag of Words (BoW) model. In BCF, a shape is decomposed into contour fragments, each of which then individually is described using a shape descriptor, and is encoded into a shape code. Finally, a compact shape representation was built by pooling shape codes in the shape. Shape classification with BCF also requires an efficient linear SVM classifier. Another work [2] proposed a skeleton-based algorithm for 2D and 3D shape retrieval. The algorithm starts by drawing circles (spheres for 3D) of increasing radius around skeletons. Since each skeleton corresponds to the center of a maximally inscribed circle (sphere), this process results in circles (spheres) that are partially inside the shape. Computing the ratio among pixels that lie within the shape, the total number of pixels was considered to distinguish shapes with similar skeletons. Complementary, in [19], a shape descriptor, named Skeleton-associated Shape Context (SSC), was proposed using combination of contour and skeleton, according to the correspondence between and associating skeletal information with a shape contour. The Bag of Features framework was applied to the SSC descriptors and finally, the shape feature vectors was fed into a linear SVM classifier to recognize the shape. This method was called Bag of Skeleton-associated Contour Parts.

Some other papers used different approaches in shape-based object recognition and classification. For example, in [20] the interaction between pattern recognition and bioinformatics, was considered to address the 2D shape classification problem. In the paper, three methods were proposed to encode a shape as a biological sequence and standard biological sequence analysis tools was employed to derive a similarity of those encodings, which then was exploited in a nearest neighbor framework. And in [21], the authors tried to apply graphical models to learn a shape representation and proposed a pipeline of shape-based object recognition. First, a Bayesian Network was used to represent the shape knowledge of a type of object. Second, an evidence accumulation inference with Bayesian Network was developed to search for the area of interest which is most likely to contain an object in an image. Finally, a spatial pyramid matching approach was used to verify the hypothesis to identify objects and to refine object locations.

Despite the enormous success of deep learning as a technique for feature learning in images, very few techniques based on deep learning have been developed for learning shape features. Specifically, no paper was found in which the researchers used CNNs to 2D shape classification, but a few endeavors have done on closely related topics including DNN-based 3D shape object recognition and sketch classification.

In 3D shape representation literature, a few works can be found in which deep leaning frameworks were utilized in 3D shape recognition. In [11], a set of algorithms and techniques was developed for learning a deep shape descriptor (DeepSD) based on the use of a multi-layer perceptron (MLP). In the paper, a heat shape descriptor (HeatSD) was developed based on point based heat kernel signature (HKS), and new definitions of Eigen-shape descriptor (ESD) and Fisher-shape descriptor (ESD) was proposed to guide the training of the MLP. In [7], an autoencoder was used for feature learning on 2D images obtained by projection of 3D shapes into 2D space, combining the global deep learning representation with local descriptor representation. Another work [22] used this method to address the problems of learning shape representation, denoising and completion in an unsupervised manner. The use of CNNs for solving object recognition tasks using RGB-D data, was presented in [23], where the possibility of making transfer learning while training CNNs was investigated. In the paper, four independent CNNs, were used for each channel, instead of using a single CNN receiving the four input channels, and these four independent CNNs were trained in a sequence, and the weights of a trained CNN was used as starting point to train the other CNNs that processed the remaining channels.

Sketch recognition task is another shape related domain where a few papers can be found that in which researchers used DNNs in sketch-based object classification. The major work on utilizing Deep CNNs for free-hand sketch recognition was Sketch-a-net [12] that aimed to exploit the unique characteristics of sketches, including multiple levels of abstraction and being sequential in nature. In the work, ensemble fusion and pre-training strategies were applied to boost the recognition performance. Another recent

attempt in this area is [13], where a deep convolutional neural network, similarly named SketchNet was proposed for sketch classification. But the main purpose of the paper was to automatically learn the shared structures that exist between sketch images and real images. In order to do this, a triplet was composed of sketch, positive and negative real image that was developed as the input of the neural network. The authors used SoftMax as lost function and ranked the results to make the positive pairs to obtain a higher score comparing over negative ones to achieve robust representation.

CONVOLUTIONAL NEURAL NETWORKS

A convolutional neural network (CNN) is a type of deep neural network (DNN) inspired by the human visual system, used for processing images. A CNN has several stages, each usually composed of two layers: the first layer does a convolution of the input image with a filter and the second layer down-samples the result of the first layer, using a pooling operation. These stages build increasingly abstract representations of the input pattern: the first might be sensitive to edges, the second to corners and intersections and so on. The idea is that these representations become both more abstract and more invariant as the pattern data goes through the CNN. The output of the last stage is usually a vector (not an image) that is fed to a multi-layer perceptron (MLP) that produces the final network output, usually a class label.

A convolutional neural network (CNN) is a type of deep neural network (DNN) inspired by the human visual system, used for processing images. A CNN has several stages, each usually composed of two layers: the first layer does a convolution of the input image with a filter and the second layer down-samples the result of the first layer, using a pooling operation. These stages build increasingly abstract representations of the input pattern: the first might be sensitive to edges, the second to corners and intersections and so on. The idea is that these representations become both more abstract and more invariant as the pattern data goes through the CNN. The output of the last stage is usually a vector (not an image) that is fed to a multi-layer perceptron (MLP) that produces the final network output, usually a class label.

Various CNN architectures were proposed to be used for object recognition. Among them LeNet [16] and AlexNet [6] have been considered as a baseline for various tasks. The former, was used to classify tiny images of hand-written digits (28×28 pixels) while the latter is somewhat complicated and has been considered as a breakthrough in object recognition on colorful medium sized images (256×256). Binary shape images have simple structures without color and texture cues, and are meaningful in very small sizes. Thus in this paper, shape images are considered as 32×32 pixels. In order to classify these images, the CNN architectures must be used that are suitable to work with this input size. Therefor in this paper, two already used CNN architectures, LeNet, MNIST-CNN [17], and a proposed architecture (for ease of reference this network is named BS-CNN stands for Binary Shape CNN) are trained to classify images of size 32×32 pixels. These architectures are described in [Table 1].

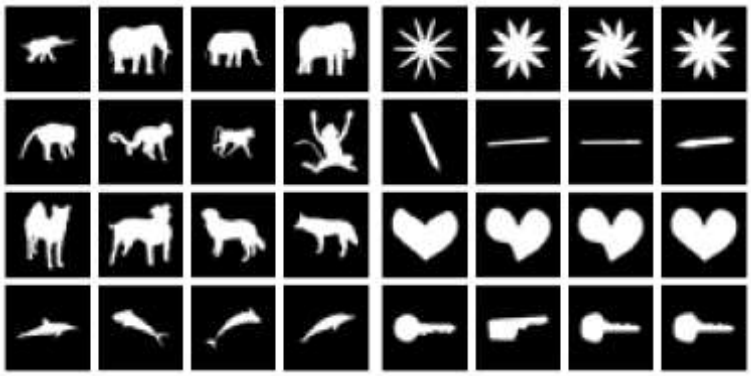
In all networks, the ReLU activation function [6] is used for each convolutional layer and MaxPooling for each Pool layer. A fully connected layer is defined as a convolutional layer with filter size of 1×1 as it is a convention in MatConvNet [24]. The final layer has 20 or 70 output units corresponding to 20 and 70 categories of Mpeg7 and Animal datasets, respectively. After all layers a SoftMax loss is placed.

EXPERIMENTS

In this section, the proposed CNN-based shape classifiers is tested on Animal [25] and Mpeg7 [15] datasets. The former consists of 2000 binary shapes of 20 animal categories with 100 shapes for each category, while the latter consists of 1400 silhouette shapes of 70 classes with 20 shapes in each class. A few samples of these datasets are depicted in [Fig. 1]. The proposed CNNs are designed for input size of 32×32 pixels but images of both datasets have variety of sizes. In order to convert them to squared scale, first, all black pixels around the shape in an image is removed, then new black pixels around the shape is added so that firstly, the resultant scale is squared and secondly, the minimum black margins to shift the

Table 1: The architecture of our CNNs

CNNs	Attributes	Layer 1		Layer 2		Layer 3		Layer 4
LeNet	Type	Conv	Pool	Conv	Pool	Conv		Conv
	Filter Size	5×5	2×2	5×5	2×2	4×4		2×2
	Stride	1	2	1	2	1		1
	Number of Filters	20	20	50	50	500		20,70
	Output Size	28×28	14×14	10×10	5×5	2×2		1×1
MNIST-CNN	Type	Conv	Pool	Conv	Pool	Conv	Pool	Conv
	Filter Size	5×5	2×2	5×5	2×2	3×3	2×2	1×1
	Stride	1	2	1	2	2	2	1
	Number of Filters	20	20	40	40	150	150	20,70
	Output Size	28×28	14×14	10×10	5×5	2×2	1×1	1×1
BS-CNN	Type	Conv	Pool	Conv		Conv		
	Filter Size	5×5	5×5	5×5		5×5		
	Stride	1	5	1		1		---
	Number of Filters	150	150	250		250		
	Output Size	28×28	9×9	1×1		1×1		



(a) (b)

Fig. 1: Sample images of datasets: (a) Animal, (b) Mpeg7

image for augmentation is considered. Then the image is resized to 32×32 pixels and each pixel's value is converted to the range [0, 1].

Data augmentation, is a method to generate extra samples from a given image. It is the process of perturbing the image by transformations that change its appearance while leaving the high-level information intact. The simplest methods of data augmentation are reflection and shifting the image pixels. For the augmentation phase of experiments, four options are considered, including no-augmentation, reflection (X2), reflection plus two random shifts (X4) and reflection plus four random shifts (X6). The reflection is done horizontally and for the random shifts, first the side of the shift is selected randomly, then after applying the shift on the image, it is flipped with the probability of 50%. These conventional data augmentation strategies will increase the training data size by a factor of 2, 4 and 6 respectively. These resultant data are randomly split into 70% train, 10% validation and 20% test sets.

The hyper-parameters of the experimented CNNs are set as follows. In all experiments, networks are trained by stochastic gradient descent with 0.9 momentum. The learning rate is set to be 0.001. The weight decay parameter is 0.0005. Initial weights are selected randomly from standard normal distribution, multiplied by 0.01. The mini-batch size is set to 100 as well as the maximum number of epochs.

The shape classification accuracies (the mean average precision) of proposed CNNs on Mpeg7 and Animal datasets, are reported in [Table 2] and [Table 3], respectively. For each experiments on a dataset with specific augmentation, top 1 and top 5 accuracies are presented.

[Table 2] and [Table 3] show the superiority of the proposed CNN architecture (BS-CNN) in comparison with the already used models, LeNet and MNIST-CNN. The proposed CNN architecture is also simpler than LeNet and MNIST-CNN. These tables also show that how augmentations lead to improve the classification accuracy on both datasets. The resultant classification accuracies can be compared with the results of the recent paper, [19], where the average classification accuracy of the proposed method, Bag of Skeleton-associated Contour Parts (BSCP), on Mpeg7 and Animal datasets were stated as 98.41% and 89.04% respectively. It is observable that the proposed BS-CNN on Mpeg7 dataset with augmentation type X4 closely compete with BSCP (97.86%), and outperforms it on Animal dataset (92.00%). BS-CNN with augmentation type X6, outperforms BSCP on both datasets (98.99% and 96.51% respectively), especially on Animal dataset. The results suggest that CNN-based classifiers with help of regular data augmentations can easily outperform current shape classification approaches. It must be noted that training BSCP, as its

Table 2: The classification results of CNNs for shape recognition on Mpeg7 dataset

MPEG7	No Augmentation		Augmentation X2		Augmentation X4		Augmentation X6	
CNN	Top 1 Accuracy	Top 5 Accuracy	Top 1 Accuracy	Top 5 Accuracy	Top1 Accuracy	Top 5 Accuracy	Top 1 Accuracy	Top 5 Accuracy
MNIST-CNN	86.43	95.00	91.43	96.96	91.16	97.86	94.70	98.69
LeNet	87.14	96.43	92.50	96.61	88.93	95.45	94.52	98.63
BS-CNN	89.64	95.71	93.75	97.86	**97.86**	99.82	**98.99**	99.76

Table 3: The classification results of CNNs for shape recognition on Animal dataset

Animal	No Augmentation		Augmentation X2		Augmentation X4		Augmentation X6	
CNN	Top 1 Accuracy	Top 5 Accuracy	Top 1 Accuracy	Top 5 Accuracy	Top 1 Accuracy	Top 5 Accuracy	Top 1 Accuracy	Top 5 Accuracy
MNIST-CNN	61.75	89.75	70.00	92.88	78.44	95.13	88.58	97.71
LeNet	66.50	89.75	72.25	93.25	74.50	94.19	87.63	97.79
BS-CNN	70.25	94.25	78.88	96.75	**92.00**	98.81	**96.51**	99.79

authors had stated, was took about 8 hours on a workstation, but training of BS-CNN, with noted hyper-parameters, takes about half an hour on a regular PC. This denotes that training of the proposed method is very fast on such a small datasets.

In above experiments, the input size of CNNs is fixed as 32×32 pixels, because firstly, binary shapes have less details than color images and they well distinguishable even in small sizes. Secondly, CNNs can work faster on small inputs as in our experiments which the time of trainings on these small datasets is in the range of a few minutes to half of an hour on the regular PC with 3.30 GHz CPU and 8 GB RAM, using CPU mode.

CONCLUSIONS

In this paper a CNN architecture is proposed to classify binary shape images with small scales. Choosing tiny input size is rational because binary shapes have less details than color images and they easily are differentiated in small sizes and the training time of CNNs can be very low. With help of a little augmentation, better results in comparison with current best results , [19], are achieved. The main focus of this paper is the architectures of CNN and other hyper parameters are set to commonly best fixed values, thus it is not hard to believe that even better results can be achieved with further investigations on architectures and hyper parameters and applying recent modifications and discoveries on deep neural networks.

CONFLICT OF INTEREST
There is no conflict of interest.

ACKNOWLEDGEMENTS
None

FINANCIAL DISCLOSURE
None

REFERENCES

[1] Goyal A, Walia E. [2014] Variants of dense descriptors and Zernike moments as features for accurate shape-based image retrieval. Signal, Image and Video Processing. 8:1273-1289.

[2] Sirin Y, Demirci MF. [2016] 2D and 3D shape retrieval using skeleton filling rate. Multimedia Tools and Applications. 1-26.

[3] Kazmi IK, You L, Zhang JJ. [2013] A survey of 2D and 3D shape descriptors. In: Computer Graphics, Imaging and Visualization (CGIV), 2013 10th International Conference, IEEE, pp. 1-10.

[4] Nanni L, Lumini A. [2013] Heterogeneous bag-of-features for object/scene recognition. Applied Soft Computing. 13, 2171-8.

[5] Ramesh, B, Xiang, C, Lee, T.H. [2015] Shape classification using invariant features and contextual information in the bag-of-words model. Pattern Recognition. 48:894-906.

[6] Krizhevsky A, Sutskever I, Hinton GE. [2012] Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp. 1097-105.

[7] Zhu Z, Wang X, Bai S, Yao C, Bai X. [2016] Deep learning representation using autoencoder for 3d shape retrieval. Neurocomputing. 204:41-50.

[8] Simonyan, K, Zisserman, A. [2014] Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[9] Côté MA, Larochelle H. [2016] An Infinite Restricted Boltzmann Machine. Neural computation.

[10] Filipe S, Alexandre LA. [2013] From the human visual system to the computational models of visual attention: a survey. Artificial Intelligence Review. 39, 1-47.

[11] Fang Y, Xie J, Dai G, Wang M, Zhu F, Xu, T, et al. [2015] 3d deep shape descriptor. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2319-2328.

[12] Yu Q, Yang Y, Liu F, Song YZ, Xiang T, Hospedales TM. [2016] Sketch-a-Net: A Deep Neural Network that Beats Humans. International Journal of Computer Vision. 1-15.

[13] Zhang H, Liu S, Zhang C, Ren W, Wang, R, Cao X. [2016] SketchNet: Sketch Classification With Web Images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1105-1113.

[14] Eitz M, Hildebrand K, Boubekeur T, Alexa M. [2011] Sketch-based image retrieval: Benchmark and bag-of-features descriptors. IEEE Transactions on Visualization and Computer Graphics. 17: 1624-1636.

[15] Latecki LJ, Lakamper R, Eckhardt T. Shape descriptors for non-rigid shapes with a single closed contour. In: Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on, [2000], IEEE, 1: 424-429.

[16] LeCu, Y, Bottou L, Bengio Y, Haffne, P. [1998] Gradient-based learning applied to document recognition. Proceedings of the IEEE. 86:2278-324.

[17] Sato, I, Nishimura, H, Yokoi, K. [2015] APAC: Augmented PAttern Classification with Neural Networks. arXiv preprint arXiv:1505.03229.

[18] Wang X, Fen, B Bai, X Liu, W, Latecki LJ. [2014] Bag of contour fragments for robust shape classification. Pattern Recognition. 47: 2116-2125.

[19] Shen W, Jiang Y, Gao W, Zeng D, Wan, X. [2016] Shape recognition by bag of skeleton-associated contour parts. Pattern Recognition Letters.

[20] Bicego M, Lovato P. [2016] A bioinformatics approach to 2D shape classification. Computer Vision and Image Understanding. 145:59-69.

[21] Wei H, Yu Q, Yang C. [2016] Shape-based object recognition via Evidence Accumulation Inference. Pattern Recognition Letters. 77:42-49.

[22] Sharma A, Grau O, Fritz M. [2016] VConv-DAE: Deep Volumetric Shape Learning Without Object Labels. arXiv preprint arXiv:1604.03755.

[23] Alexandre LA. [2016] 3D object recognition using convolutional neural networks with transfer learning between input channels. In: Intelligent Autonomous Systems 13: 889-898.

[24] Vedaldi A, Lenc K. [2015] Matconvnet: Convolutional neural networks for matlab. In: Proceedings of the 23rd ACM international conference on Multimedia, ACM, pp. 689-92.

[25] Bai X, Liu W, Tu Z. [2009] Integrating contour and skeleton for shape classification. In: IEEE Workshop on NORDIA