Image Annotation using CGS-CNN with Limited Training Example

Pallavi D. Chaudhari [1], Dr. Nitin N. Patil [2]
[1]*P.G. Student*
Department of Computer Engineering SES's R. C. Patel Institute of Technology, Shirpur
[2] *Head & Associate Professor*
Department of Computer Engineering SES's R. C. Patel Institute of Technology, Shirpur
(E-mail: pallavidc14@gmail.com, er.nitinpatil@gmail.com*)*

Abstract— Automatic image annotation is a challenging issue in the field of image retrieval. It can be utilized to encourage semantic search in extensive image database. Several techniques have been proposed for image annotation in the most recent decade that gives sensible performance on standard datasets. The drawback of this techniques is that it requires an immense quantity of training images with clear and complete annotation for a decent model for tag prediction. In this work, we address this limitation by proposing the real-time object detection system that combines selective search to extract possible objects using a region proposal method. We also integrate a canny edge detection to identify a wide range of edges in images.

Keywords—Automatic image annotation; Multi-label image; object detection; image retrieval; Tag ranking.

I. INTRODUCTION

Recently, due to easy accessibility and low value of high-resolution digital cameras digital image collections growing speedily. To retrieve images accurately from these wide collections of digital images has become a valuable research issue. To retrieval for images relevant to a query some image processing methods are used for features extraction. The automatic image annotation captures semantic features with machine learning techniques. The objective of image annotation is to automatically annotate an image with suitable keywords that return its visual contents. It does a major role in bridging the semantic gap between low-level features and high-level semantic images contents. Most of the studies consider image annotation as a multilabel classification problem in multiple objects images [1, 2].

Tag ranking intends to study a ranking function that places relevant tags in front of the irrelevant ones. In the simplest type, it learns a scoring function that allocates larger values to the relevant tags than to irrelevant tags. However, several algorithms have been developed for tag ranking, but they tend to execute poorly when the number of training images is limited as compared to the number of tags [3].

The visual content to be analyzed in the automatic image annotation depends on instances of objects. Object detection is an approach to detect the objects from the given image with a specific measure or method. Recently, massive deep detection techniques are proposed with the evolution of deep learning in object detection. In object detection, the present mainstream deep learning models can be categorized into two major parts as the model based on region proposal and the model based on regression. Unlike image classification and detection needs localizing (likely many) objects within the image. One approach frames localization as a regression problem [4].

We noticed that many algorithms presented for tag ranking but they still perform poorly as the number of training images are limited in compared to the number of tags. In many studies consider image annotation approach as a multi-label classification problem. The key disadvantage of this problems is that it needs a large amount of training images with clear and complete annotations. In this system we address this problem with developing a real-time object detection system with image annotation. In our system, we describe object detection as a regression problem to the correlates a variety of bounding boxes. Additionally, for every expected box the net outputs a dependence score of probably this box accommodates object. This can be quite completely different from conventional approaches that score features inside predefined boxes, and has the advantage to demonstrate detection of objects in an efficient and compact approach and enhance the accuracy of our system.

II. RELATED WORK

The tags quality plays a key role in social image retrieval. Recent years to deal with the tag quality problems have witnessed a lot of emerging studies. In this section, we analyze and summarized some representative techniques that are closely related to the approach presented in this work. In recent years various techniques have been proposed for automatic image annotation. Generally automatic image annotation can be considered as an intermediary problem for a general web image retrieval task. Nearly, most of these techniques aim to model the probabilistic relationship between images and tags. Although, generating extremely accurate annotation outcomes remains an unsolved long-term challenge. Tag refinement is an alternative approach to study instead of auto-annotation, which objective is to model the relevance of the associated tags to an image [5].

S. Liu et al. uses the Kernel Density Estimation (KDE) in order to compute relevance scores for differentiate tags, and carry out a randomwalk to further enhance the performance of

tag ranking by exploring the correlation among tags. J. Zhuang et al. proposed a two-view tag weighting method that strongly utilize to both the correlation between tags and the dependence among visual features and tags. A max-margin riffled independence model is proposed by T. Lan et al. for image tag ranking. As the survey of existing methods for tag ranking lean to execute poorly when tag space is huge and the number of training images is limited [5, 6, 7].

Object detection techniques acquired in the manner of region selection + feature extraction + classification is established on deep learning, the region selection can be performed as stated in to some strategy, the classification can be recognized by traditional the special neural network or SVM and the feature extraction can be performed by the convolutional neural network. The rapid representative methods of deep learning applied in object detection are DNN and Overfeat. DNN object detection has developed two subnetworks that contain the regression subnetwork for location and the classification subnetwork for recognition. Initially, for classification DNN is the deep neural network. Suppose, in the rear the softmax layer is replaced with regression layer then DNN can act as the regression subnetwork and can achieve the object detection task as it combined with the classification subnetwork. The Overfeat is developed by LeCun's et al., that extracts features with the enhanced deep convolutional model AlexNet, allowing the offset and slide window to understand the goal of object classification by exploitation images of varied scales and locate objects by merging the regression network, therefore achieving the object detection [8, 9].

Carreira et al. developed Constrained Parametric Min-Cuts for Automatic Object Segmentation and Endres et al. developed Category Independent Object Proposals. Both proposed systems create a collection of class independent object hypotheses using segmentation. Both systems create several foregrounds or background segmentations, learn to guess the probability that a foreground segment is a complete object, and utilize this to segments ranking. Those show a favorable capability to correctly describe objects within images. To identifying good regions both systems depend on a single strong algorithm. They acquire various locations by utilizing several randomly initialized foreground and background seeds. A selective search approach, in contrast completely deals with various image conditions by using a variety of grouping criteria and distinct representations [10, 11, 12].

III. METHODOLOGY

In this section, we introduce the proposed system for object detection and tag ranking which is specifically constituted for a large tag space with a limited number of training images. Our CGS-CNN system is based on regression approach. Our proposed system predicts bounding boxes utilizing dimension clusters as anchor boxes. The system predicts 4 arranges for each bounding box, tx, ty, tw, th. On the off chance that the cell is counterbalanced from the upper left corner of the picture by (cx, cy) and the bounding box with the width and tallness pw, ph. Each case predicts the classes the bounding box may contain utilizing multilabel grouping. We don't utilize a softmax as we have discovered it is pointless for good execution, rather we just utilize independent logistic classifiers. This definition encourages when we move to more perplexing Datasets like the ESPN Dataset. In this dataset, there are many overlapping marks. Utilizing softmax forces the supposition that each crate has precisely one class which is frequently not the situation. A multilabel approach better models the information. Our Proposed system predicts boxes at 3 unique scales and extricates features from those scales utilizing a comparative idea to feature pyramid networks. From our base element extractor, we include a few convolutional layers. At last of these predicts a 3-d tensor encoding bounding box, objectness, and class predictions.

In our investigations with COCO we predict 3 boxes at each scale so the tensor is N × N × [3 ∗ (4 + 1 + 80)] for the 4 bounding box counterbalances, 1 objectness expectation and 80 class predictions.

A. *Canny Edge Detection*

In our system, we use the Canny edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images, and Additionally extracts useful structural information from distinct vision objects and reduces the amount of data to be processed.

Canny edge detection algorithm

STEP I: Smooth the image with a Gaussian filter to reduce noise and unwanted details and textures.

$$g(m,n) = G\sigma(m,n) * f(m,n) \quad (1)$$

Where, $G\sigma = \frac{1}{\sqrt{2\pi\sigma^2}}\exp(-\frac{m^2+\ n^2}{2\sigma^2})$

STEP II: Compute the gradient of $g(m,n)$ using any of the gradient operatiors (Roberts, Sobel, Prewitt, etc) to get:

$$M(m,n) = \sqrt{g_m^2(\mathrm{m,n}) + \ g_n^2(\mathrm{m,n})} \quad (2)$$

and

$$\theta(m,n) = \tan^{-1}[g_n\ (m,n)/g_m\,(m,n)]$$

STEP III: Threshold M

$$M_T\ (m,n) = \begin{cases} M\ if\ M(m,n) > T \\ 0 \quad otherwise \end{cases}$$

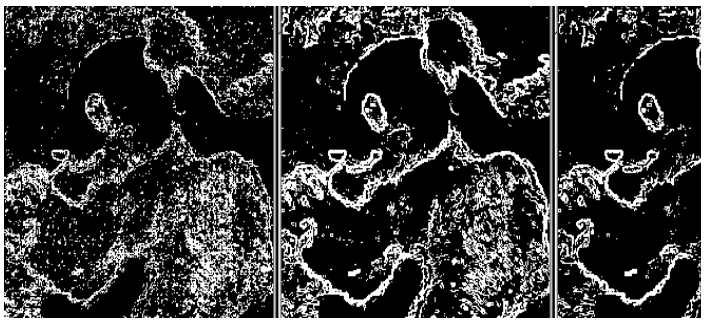
where T is so chosen that all edge elements are put up while most of the noise is suppressed.

STEP IV: Suppress non-maxima pixels in the edges in M_T obtained above to thin the edge ridges (as the edges might have been broadened in step 1). To do that, check to see whether every non-zero $M_T\ (m,n)$ is greater than its two neighbors along the gradient direction $\theta(m,n)$. If so, put $M_T\ (m,n)$ unchanged, otherwise, set it to value 0.

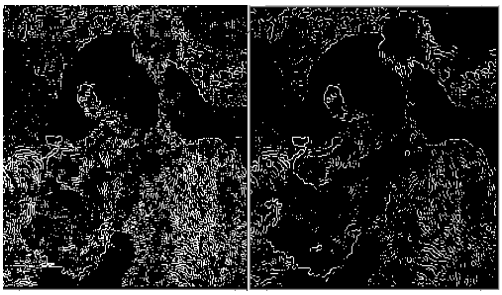
STEP V: Threshold the previous result by two different thresholds τ_1 and τ_2 (where $\tau_1 < \tau_2$) to obtain two binary images T_1 and T_2. Note that T_2 with greater τ_2 has low noise and fewer false edges although greater gaps between edge segments, when compared to T_1 with smaller τ_1.

STEP VI: Link edge segments in T_2 to form continuous edges. To do so, trace each segment in T_2 to its end and then search its neighbors in T_1 to find any edge segment in T_1 to bridge the gap untill reaching another edge segment in T_2.

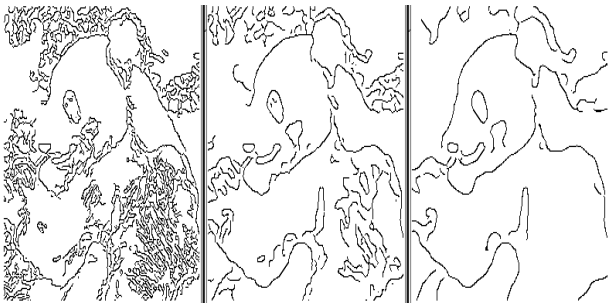
(a) Original Image



(b) Edge detection by gradient operators (Roberts, Sobel and Prewitt)



(c) Edge detection by LoG and DoG



(d) Edge detection by Canny method ($\sigma = 1,2,3$, $\tau_1 = 0.3, \tau_1 = 0.7$)

Fig 1: Canny Edge Detection of the Image

Fig. 1 shows the edge detection of the image with Canny. As shown Fig. 1(a) is the original given image, Fig. 1(b) is the Edge detection by gradient operators Roberts, Sobel and Prewitt, Fig. 1(c) shows edge detection by LoG and DoG, and Fig. 1(d) detects the edges by Canny method. In above algorithm we used Mathematical equations from [13] proposed by J. Canny et al. where σ is the standard deviation of Gaussian function. In equation (1) First derivation of two-dimensional Gaussian G is $G\sigma$ in the same direction σ. Also g_m and g_n are the results which are effects of filter f_m and f_n on original image in equation (2).

B. Gaussian Blur

In our proposed system we utilized a Gaussian blur to reduce image noise and reduce detail. The noise reduction is done with the result of blurring an image by a Gaussian function. The visual effect of this blurring to the given image is a smooth blur resembling that of viewing the image across a translucent screen and distinctly varied from the bokeh effect generated by an out-of-focus lens or the shadow of an object below usual illumination. We utilized Gaussian smoothing in the pre-processing stage in our system in order to enhance image structures at unlike scales. Mathematically, applying a Gaussian blur to an image in our system is the as like as convolving the image with a Gaussian function. The Fourier transform of a Gaussian is another Gaussian and applying a Gaussian blur has the result of reducing the image's high-frequency components. Since Gaussian blur is acting as a low pass filter in our CGS-CNN system.

C. Selective Search

In our system the possible object locations to be use in object recognition for a query image is generated by selective search method. A selective search algorithm has following capabilities and presents a various diversification strategies to deal with multiple image conditions as possible.

- *Capture All Scales*: Objects can exist at any scale in the image. Additionally, some objects have little clear-cut boundaries than other objects. consequently, in selective search all object scales have to be taken into

account. This is most obviously acquired by utilizing a hierarchical algorithm.

- *Diversification*: There is no isolated appropriate approach to group regions together. In some conditions of images regions may form an object because of only colour, only texture, or because parts are enclosed. Additionally, lighting conditions such as the colour of the light and shading may affect how regions form an object. Hence instead of an isolated approach which works well in most conditions, we want to have a diverse set of approaches to deal with all cases.
- *Fast to Compute*: The objective of utilizing selective search in our system is to provide a set of possible object locations for use in a practical object recognition framework. This algorithm is reasonably fast as the production of this set should not become a computational restriction.

Algorithm of Selective Search

STEP I: Generate initial sub-segmentation. Goal is to generate several regions, each of which belongs to only single object.

Using the method described by Felzenszwalb et al. from week 1 works well.

STEP II: Recursively combine similar regions into larger ones.

Greedy algorithm:

1. From set of regions, take up two that are most similar.
2. Combine them into a single, larger region.
3. Repeat until only one region remains.

STEP III: Recursively combine similar regions into larger ones.

STEP IV: Use the generated regions to produce candidate object locations.

D. Object Detection

In Previous work on object detection classifiers are reused to perform detection. Instead our system, frame object detection as a regression problem to spatially differentiated bounding boxes and corresponding class probabilities. A single neural network predicts class probabilities and bounding boxes directly from full images in one estimation. It causes the entire detection pipeline is a single network and it can be optimized end-to-end directly on detection performance. Our proposed system resizes an input image to 448×448. In this run to a single convolutional network on an image, and threshold to the resulting detections by the model's confidence. A single convolutional network simultaneously detects multiple bounding boxes and class probabilities for those bounding boxes. This directly optimizes detection performance as it trains on full images. Our system is extremely fast as it frames detection as a regression problem and do not need a complex pipeline.

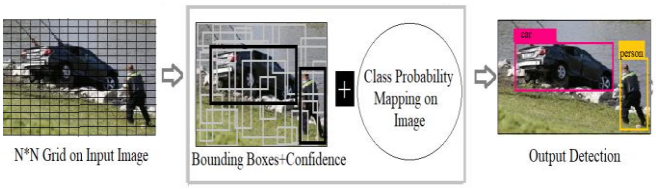


Fig. 2: The Operation of Object Detection in Proposed System

Above Fig. 2 shows the operation of our proposed system as a regression problem. System divides the input image into an $N \times N$ grid. If the center of an object falls into a grid cell then that grid cell is responsible for detecting that object. In image for every grid cell predicts B bounding boxes, confidence for those boxes, and C class probabilities. These predictions are encoded as an $N \times N \times (B * 5 + C)$ tensor toolbar.

E. Algorithm of CGS-CNN System

Algorithm of CGS-CNN System

STEP I: Take an input of the image

STEP II: Apply Canny edge detection and Gaussian blur on input image to detect the edges with removal of noise

STEP III: Image with edges detected pass to selective search algorithm to extract possible object regions

STEP IV: Object region extracted image pass as a input to ConvNet which in turns generates the Regions of Interest

STEP V: Divide the image into various regions and consider each region as a separate image

STEP VI: Generate initial sub segmentations so that to get multiple regions

STEP VII: The technique then combines the similar regions to form a larger region based on color similarity, size similarity, texture similarity and shape compatibility

STEP VIII: Pass all these regions i.e. images to the CNN and classify them into different classes.

STEP IX: As each region divided into its corresponding class then combine all these regions to get the original image with the detected objects.

STEP X: Apply region proposal network on these feature maps that returns the object proposals along with their objectness score.

STEP XI: Use of a softmax layer on top of the fully connected network to output classes, with the softmax layer, parallelly use of a linear regression layer to output bounding box coordinates for predicted classes

STEP XII: Obtain the output of image annotation with bounding box prediction

IV. EXPERIMENTAL RESULTS

In this section, we describe our experimental setup, including image datasets and feature extraction. We then presented three sets of experiments to calculate the effectiveness of the proposed system, where the first experiment examines the performance of image annotation using training

images with missing tags and in second, we evaluate the sensitivity of the proposed algorithm to parameter λ. To evaluate the proposed system, we perform extensive experiments on IARTC-12 datasets for image annotation and sensitivity.

A. *Automatic Image Annotation*

In this experiment, we examined the performance of proposed system when training images are partially annotated. We randomly selected 20% of the assigned tags for training images. This setting allows to test sensitivity of the proposed system to the missing tags. We perform the experiment on IAPRTC-12 dataset. The result for average precision of IAPRTC-12 dataset is as shown in Fig. 3.

TABLE I.

COMPARISION OF AUTOMATIC IMAGE ANNOTATION PERFORMANCE

Sr. No.	Average Precision@K on IAPRTC-12	Existing System	Proposed System
1	K = 1	37	41
2	K = 2	28	30
3	K = 3	22	24
4	K = 4	20	21

Fig. 3: Automatic Image Annotation performance on IAPRTC-12 dataset with incomplete image tags

It is not surprising to observe that annotation performance of proposed system drops as the number of observed annotations goes on decreases, indicates that the missing annotations could mainly affect the annotation performance. This result indicates that the proposed method is more effective in handling missing tags. Fig. 3 provides examples of annotations generated by proposed system and existing system for the IAPRTC-12 dataset when only 20% of the assigned tags are observed for every training image. The TABLE I reports the comparison of automatic image annotation performance of proposed system with existing system on IAPRTC-12 dataset. The TABLE I indicates the Average precision at K when 20% of the assigned tags are observed. These examples further confirm the advantage of using the proposed system for automatic image annotation when the training images are equipped with incomplete tags.

B. *Sensitivity*

In this experiment, we estimated the sensitivity of the proposed method to parameter λ. To examine the sensitivity, we use the IAPRTC-12 dataset. Generally, a larger λ will cause to a higher regularization capacity and as a sequence, a greater bias and a smaller variance for the final solution.

TABLE II.

SENSITIVITY OF PROPOSED SYSTEM TO PARAMETER λ

Sr. No.	λ	Average Precision @K K = 1	Average Precision @K K = 4	Average Precision @K K = 7	Average Precisio n@K K = 10
1	0.01	45	35	28	28
2	0.1	48	37	35	25
3	1	52	46	39	22
4	10	44	42	33	18
5	100	35	32	30	14

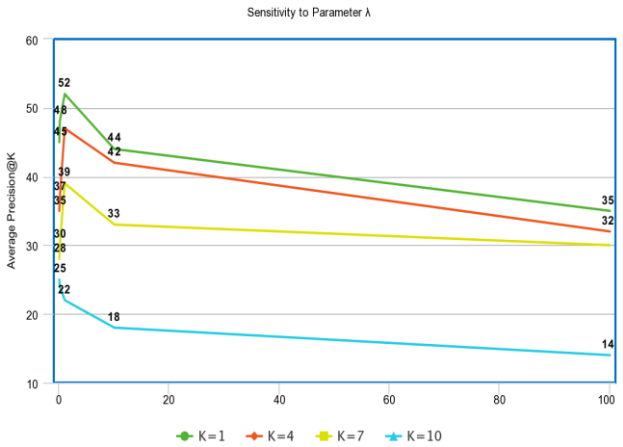


Fig. 4: Average precision of the proposed system on IAPRTC-12 with varied λ

The TABLE II shows the Average precision@K for which, we consider the values as K = 1, K = 4, K = 7 and K = 10. The table indicates the sensitivity of our proposed system to the parameter λ. We observed that when λ = 1 then performance of proposed system is high.

The Fig. 4 indicates Average precision of the proposed system with varied λ on IAPRTC-12 dataset. It shows the estimation of sensitivity of proposed system for K, where we consider the values of K as 1, 4, 7 and 10. In order to see how the parameter affects the annotation performance, we perform the experiment by varying λ from 0.01 to 100 and measure average precision for the proposed system. We observe that the proposed method yields the best performance when λ is around 1.

V. CONCLUSION

In this work, we have proposed the CGS-CNN system for image annotation with the limited number of training images.

Our real-time object detection system has many benefits over the classifier-based system. It makes easy and effective object recognition as we utilize selective search. Our model is extremely fast, as at test time it considers the complete image and it predicts with one network only where method like R-CNN requires thousands for a single image. At long last, we propose an algorithm to train on the labelled images. Utilizing this strategy, we train the algorithm on the COCO recognition dataset furthermore, the ESPN Image dataset. Our algorithm permits to foresee recognitions for classes that don't have marked information. In future, the proposed system to the image annotation problem may enhanced for real-time detection at high FPS with improved accuracy.

REFERENCES

[1] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid, "Tagprop: Discriminative Metric Learning in Nearest Neighbor Models for Image Auto-annotation," in IEEE Int. Conf. on Computer Vision, 2009, pp. 309– 316. L.

[2] S. Zhang, J. Huang, Y. Huang, Y. Yu, H. Li, and D. Metaxas, "Automatic Image Aannotation using Group Sparsity," in IEEE Int. Conf. on Computer Vision and Pattern Recognition, 2010, pp. 3312–3319.

[3] S. Feng, Z. Feng, and Rong Jin, "Learning to Rank Image Tags with Limited Training Example," IEEE Trans. Image Processing, VOL. 24, NO. 4, APRIL 2015

[4] R. Girshick, J. Donahue, T. Darrell, et al, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580-587.

[5] J. Zhuang and S. C. H. Hoi, "A Two-view Learning Approach for Image Tag Ranking," in Proc. 4th ACM Int. Conf. WSDM, 2011, pp. 625–634.

[6] D. Liu, X. Hua, L. Yang, M. Wang, and H. Zhang, "Tag Ranking," in WWW, 2009, pp. 351–360.

[7] T. Lan and G. Mori, "A Max-margin Riffled Independence Model for Image Tag Ranking," in IEEE Int. Conf. on Computer Vision and Pattern Recognition, 2013, pp. 3103–3110.

[8] C. Szegedy, A. Toshev, D. Erhan, "Deep Neural Networks for Object Detection," Advances in Neural Information Processing Systems, 2013, pp. 2553-2561.

[9] P. Sermanet, D. Eigen, X. Zhang, et al, "Overfeat: Integrated recognition, localization and Detection using Convolutional Networks," ICLR, 2014.

[10] Joao Carreira and Cristian Sminchisescu, "Constrained Parametric Min-Cuts for Automatic Object Segmentation", in IEEE Int. Conf. on Computer Vision and Pattern Recognition, 2013.

[11] J.R.R. Uijlings, "Selective Search for Object Recognition," IJCV, 2012.

[12] P. Chaudhari, N. Patil, "A Review of Image Annotation Methods," International Journal of Modern Trends in Engineering and Research, vol. 3, Issue.1, 2014.

[13] J. Canny, "A Computational Approach to Edge Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. pami-8, no. 6, November 1986.