Bayesian Approach to Photo Time-Stamp Recognition

Asif Shahab, Faisal Shafait, Andreas Dengel
German Research Centre for Artificial Intelligence (DFKI)
Kaiserslautern, Germany
{*Asif.Shahab,Faisal.Shafait,Andreas.Dengel*}*@dfki.de*

***Abstract*—Time-stamps and URLs overlaid artificially on images add useful meta information which can be used for automatic indexing of images and videos. In this paper, we propose a method based on an attention-based model of visual saliency to extract overlaid text and time-stamps that are rendered on images. Our model of visual saliency is based on a Bayesian framework and works very well for the task of time-stamp detection and segmentation as is evident by overall object recall of 80% and precision of 70%. Our method produces a clean text segmented binarized image, which can be used for recognition directly by an OCR system. Furthermore, our technique is robust against variation of font styles and color of time-stamp and overlaid text.**

***Keywords*-overlaid text detection/recognition, photo time-stamp detection/recognition, visual saliency, Bayesian model for text detection**

I. Introduction

Extraction of text occurring naturally *(Scene text)* or artificially *(Overlaid text)* in an image has been the focus of research for many years. Scene text occurs naturally in an image and is difficult to separate because of illumination problems, perspective distortions and occlusion. Overlaid or artificial text is usually added by cameras in form of time-stamps and URLs or by image editing software on top of the image. They are usually upright and are added with readability in mind. Though time-stamps are usually put at a distinct location, in distinct color and font, the problem of extracting these time-stamps from images is still a complex one. Firstly, because it can appear on highly textured backgrounds making it difficult to separate from the background. Secondly, the high dimensionality of colored images and the variety of fonts, colors and formats time-stamps can occur in an image makes it a challenging problem.

Several approaches have been reported in literature to solve the problem of text extraction from images. These approaches can be classified into region based and texture based methods [1].

Region based methods use connected components or vertical and horizontal edges and merge them together based on some rules exploiting the geometric properties of text. These methods work on the assumption that color of text does not change and is considerably different from the background color [2] [3] [4]. They are generally faster and work well for simple backgrounds but they are sensitive to noise [5].

Texture based methods use the textural properties to separate text from the background. In order to extract the textural features they use range of frequency domain techniques like Gabor filters, FFT, DCT, Wavelets, spatial variance etc. Subsequently, they use machine learning algorithms such as SVM, AdaBoost and MLP to train a text finder [6] [7]. Pan et al. [8] recently proposed a hybrid system for scene text detection which uses a combination of texture and connected component based method and uses a CRF model to filter non-text components. These algorithms are generally slow because of high computational complexity.

Li et al. [9] [10] and Chen et al. [11] proposed systems for time-stamp detection and recognition based on time-stamp fonts template and skeleton matching. Set of templates are created for a variety of time-stamps and Sobel operators on Red and Green color channels and set of morphological operators (*close, open*) are applied for rough segmentation of image and reduce the search space for skeleton matching. The major limitation of the system is that of number of templates required to accommodate all variety of fonts and styles time-stamp can occur in. Further, the red and green channels used by Li et al. for the extraction of sobel edges limits the system capabilities to work with range of colors time-stamps can take.

In this paper we propose a system for overlaid text extraction based on attention based models of visual saliency. Since overlaid text and time-stamps are usually added with the intention of readability and thus respond well to attention based models. We have applied Bayesian framework tuned by time-stamps location based Bayesian prior learned independently from training images to calculate saliency for each pixels.

We explain our technique for saliency evaluation and time-stamps segmentation in Section 2. We report our experimental results in Section 3 and conclude the paper in Section 4.

II. Proposed Method

Our probabilistic framework for time-stamps detection is inspired by the visual saliency model for object search and contextual guidance by Torralba et al. [12]. Such a model gives for each image location, the probability of finding an object, in our case time-stamps, by integrating global and local image information using task constraints.

This is in contrast to other visual saliency models such as Itti's [13] where several image features (color, contrast, orientations) are combined to give saliency values for each image location. Torralba's model defines saliency in a Bayesian framework allowing integration of task constraints as Bayesian priors and thus can be tuned to search for specific objects in an image.

In a Bayesian framework, the probability of finding an object $p(O = 1, X|L, G)$ at a location $X = (x, y)$ given the set of local measurements $L(X)$ and a set of global features G can be expressed by:

$$p(O = 1, X|L, G)$$
$$= \frac{1}{p(L|G)} p(L|O = 1, X, G) p(X|O = 1, G) p(O = 1|G) \quad (1)$$

Different factors in Equation 1 can be explained as follows:

1) The first term, $1/p(L|G)$, is the bottom up saliency factor that represent the inverse of probability of finding local measurements in an image which is an integral part of a Bayesian framework.
2) The second term, $p(L|O = 1, X, G)$, represents the top-down knowledge of target appearance and how it contributes to the object search [12]. For the case of time-stamps, we assumed this probability factor to be uniform, since we wanted our model to be independent of time-stamp appearance.
3) The third term, $p(X|O = 1, G)$, provides the context based information and serves as a Bayesian prior. This factor can be effectively learned for the case of time-stamps, as time-stamps are usually placed at distinct image locations mostly at one of the four corners of the image. This is also independent of the Global image features and thus can be reduced to $p(X|O = 1)$
4) The fourth term, $p(O = 1|G)$, represents the probability of finding an object(time-stamp) in the scene. Since, time-stamps are put artificially by the camera and do not depend on the scene contents, we assume this probability to be uniform as well.

Hence, the final model of saliency can be derived from Equation 1 in terms of bottom up saliency factor and context based Bayesian prior:

$$S(X) = \frac{1}{p(L|G)} p(X|O = 1) \quad (2)$$

A. Saliency Factor Estimation,

We used steerable pyramid [14] filters tuned to six orientations and four scales to generate local image features as in [12]. Raw RGB channels are fed to the bank of filters to generate a set of $(6 * 4 * 3 = 72)$ features, L, for each image location (x, y). Saliency estimation requires estimating the distribution of local features in the image. We used multivariate Gaussian distribution in contrast to the multivariate power exponential distribution used by Torralba et al. [12]. The multivariate power exponential accounts for the long tail of distribution but we are only interested in finding pixel locations having highest saliency as given by Equation 2, which can equally be estimated by a Gaussian distribution using:

$$log\ p(L) = log\ k - 1/2[(L - \mu)^t \Sigma^{-1} (L - \mu)] \quad (3)$$

where k is the normalization constant, μ is the mean of each of the 72 features for all the image locations, Σ is the covariance matrix of local image features. We used maximum likelihood criteria to estimate the Gaussian distribution parameters μ and Σ. We chose to ignore the normalization constant, as this will not affect the overall maximization of saliency. Based on these distribution parameters, the probability of local measurements in an image I can be approximated by:

$$p(L|G) \approx p(L|\mu(I), \Sigma(I)) \quad (4)$$

B. Bayesian Prior Estimation

For estimating the Bayesian prior, we prepared a training set consisting of approximately 80 images with marked time-stamps bounding boxes. We then calculate the histogram of occurrence of time-stamp at a particular image location scaled by the size of the dataset. We chose to use relative image co-ordinates X=[0,1], Y=[0,1] for image locations in order to make our calculations independent of image sizes. This Bayesian prior serves as a representative of time-stamps locations in image. We found out that the training sample size of 80 images is sufficient as time-stamps usually occur on one of the four corners of the image at almost the same location.

C. Saliency Estimation, S(X)

As shown earlier, saliency for a given image location can be computed by Equation 2. The Bayesian priors in our case are learned independently of the local image features and thus we combine the two results using the weighting factor γ using the following formula:

$$S(X) = p(L|G)^{-\gamma} p(X|O = 1)^{1-\gamma} \quad (5)$$

The parameter γ is fixed at 0.05 as suggested by Torralba [12]. The saliency estimates for each pixel is then classified into top 1%, 2% and 3% bins based on the saliency value histogram. The saliency estimates are shown in Figure 1 that clearly shows our saliency model is quite effective in maximizing the saliency of time-stamp pixels.

D. Time-stamp Segmentation

We found in our experiments that time-stamps are usually covered by top 3% of the pixels in saliency estimates as shown in Figure 1. We use this as a threshold and find all the regions of interest in the image consisting of pixels with

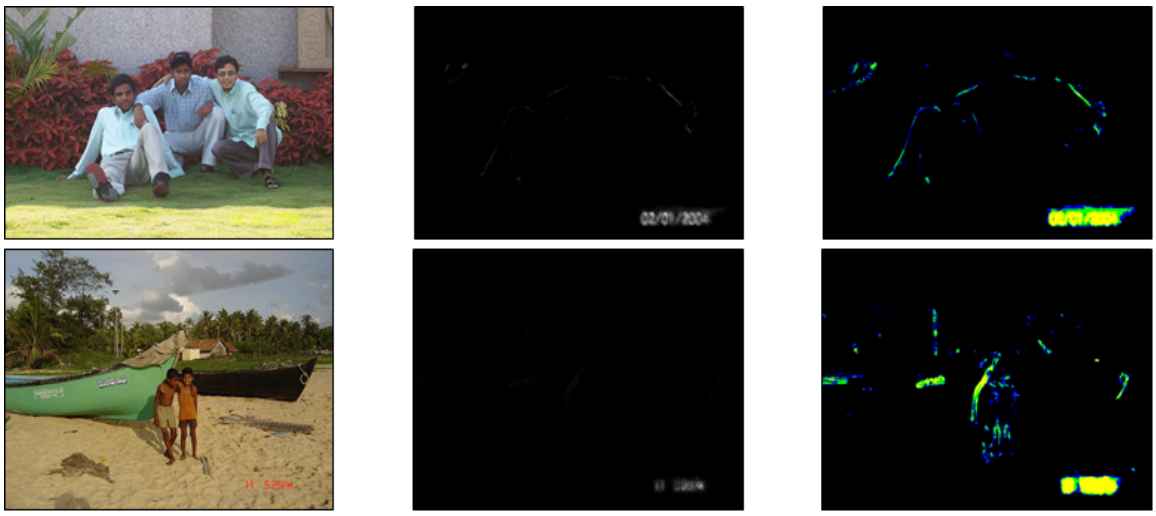


Figure 1: Original Image (left), Saliency map (middle), Saliency map (right) with top 1% pixels painted as yellow, top 2% as green and top 3% as blue

saliency values in top 3% using connected components. For each of these regions we perform the following steps:

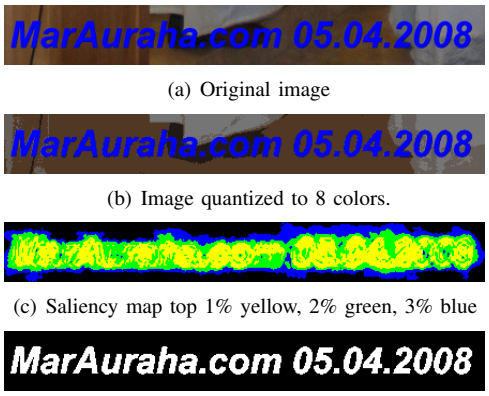
1) We crop the region in the original image and quantize it to 8 colors [15]. Since time-stamps are usually put by the camera in one distinctive color, color quantization should keep the color of time-stamps intact while reducing the number of colors in the background as shown in Figure 2b.
2) We process each color as a candidate for time-stamp color in the given image region. The color belongs to a time-stamps if the pixels belonging to this color overlaps by at least 70% with the pixels belonging to top 1% salient pixels in the given region. This threshold is set empirically.
3) We use the colors selected in the previous step as the binarization threshold and include all the pixels covered by the given color in the selected region as foreground pixels. Result is shown in Fig. 2d.

All candidate image regions are merged into one image to generate a binarized image.

E. Post-Processing

The binarized images obtained after text segmentation may contain some background noise. We remove these noisy elements and calculate time-stamps location by following steps:

1) Find connected components in the binarized image.
2) Merge horizontally and vertically adjacent connected components and grow the bounding boxes till no more merging is possible.
3) Filter expanded bounding boxes with width w and height h using the formula: $abs(log_2(w/h)) > 2$. This will filter all the bounding boxes which do not have aspect ratio characteristic of a text line.



(a) Original image

(b) Image quantized to 8 colors.

(c) Saliency map top 1% yellow, 2% green, 3% blue

(d) Binarization

Figure 2: Text segmentation for image region

The final bounding boxes are shown in the Figure 3 and marked with a red outline.

III. Experiments and Results

A. Dataset

We collected images containing time-stamps and URLs from a variety of web sources. Our dataset consists of 275 images containing time-stamps and URLs in different font styles, color, orientation and locations. We prepared ground truth for these images. Our ground truth consists of text bounding boxes and ASCII text. The training set consisting of 80 images is used to learn the Bayesian prior as described in Section 2B. Some of the sample images from the dataset and the text location and segmentation results are shown in Figure 3. As can be seen from the results, the segmented image can be directly used by the OCR system for text recognition.

Figure 3: Text Location, Segmentation Results

B. Text Localization Evaluation

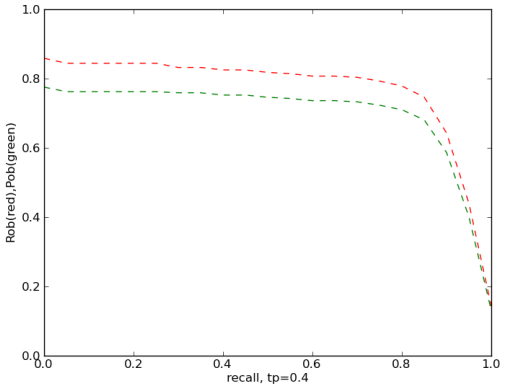
We used the approach proposed by Wolf et al. [16] based on object Count and Area Graphs for the evaluation of our text localization algorithm. This evaluation approach illustrates the performance of a detection algorithm by graphs showing object recall and precision depending on the constraints put on the detection quality. The evaluation scheme uses two quality constraints t_r and t_p on the area recall and precision of the ground truth and detected rectangles. A ground truth rectangle G_i matches a detected rectangle D_j if area recall and precision are higher than respective constraints i.e.

$$\sigma_{i,j} = R_{AR}(G_i, D_j) > t_r \text{ and } \tau_{i,j} = P_{AR}(G_i, D_j) > t_p$$

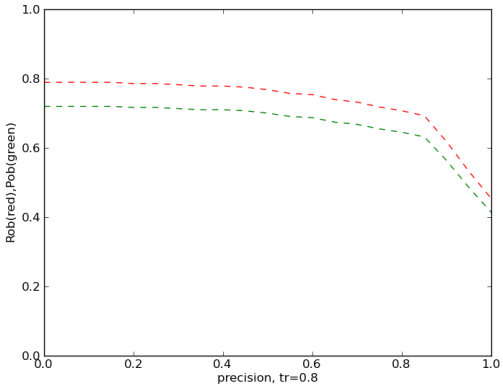
where,

$$\sigma_{i,j} = \frac{Area(G_i \cap D_i)}{Area(G_i)} \text{ and } \tau_{i,j} = \frac{Area(G_i \cap D_i)}{Area(D_i)}$$

Figure 4a shows text bounding box recall and precision graph by varying the value of recall threshold t_r, and precission threshold t_p, fixed at 0.4 as in [16]. Figure 4b shows recall and precision graph by varying the value of precision threshold t_p, and recall threshold t_r, fixed at 0.8 as in [16]. The precision threshold of 0.4 might seem a bit low but consider the fact that the area of a rectangle changes squarely with its side lengths [16]. Figure 4a and 4b clearly shows that object recall and precision are fairly constant at around 80% and 70% over the large range of quality constraints t_r and t_p. Figure 4b also shows that the majority of detection are with very high precision as time-



(a) Varying Recall Quality Constraint, tp = 0.4



(b) Varying Precision Quality Constraint, tr=0.8

Figure 4: Time-stamps Recall and Precision over range of quality constraint.

stamps recall only drops beyond 80% of precision quality constraint.

C. Text Segmentation Evaluation

We used our text-segmented images as input to different OCR systems. The recognition error rates for Omnipage, OCRopus and Tesseract are shown in Table I. The error rate of 32% for Omnipage is the best. However, it is to be noted that most of the fonts that occur in time-stamps are not standard fonts that occur in every day document images. These OCR systems are mainly trained for document layouts and fonts. We also observed that combining the results from different OCR systems by some sort of text voting scheme may decrease the overall error rate.

OCR System	GT Chars	Recog. Chars	Errors	Error Rate
Omnipage	3294	2400	1054	0.32
Tesseract	3294	1933	2051	0.62
OCRopus	3294	1988	2319	0.70
WeOCR+Tesseract	3294	1796	3300	1.00

Table I: OCR evaluation results on our text-segmented images and comparison with WeOCR

For comparison, we also evaluated the WeOCR scene text recognition system [17] with our dataset. This system recognizes text region in images and uses Tesseract for recognition. The experiment results are shown in Table I. The overall error rate for WeOCR is 100% because of many false positives resulting in noisy character recognition, whereas our segmentation algorithm results in an error rate of 62% using the Tesseract system.

IV. Conclusion

In this paper, we have proposed a Bayesian framework for the detection of time-stamps and overlaid text in image. The saliency model has proved to be quite effective in locating the time-stamps as seen in Figure 1. Segmentation results clearly show that the technique is robust against variation of font styles, orientation, color and backgrounds as shown in Figure 3. Our experimental results show that the segmentation algorithm can be directly used by the OCR system for recognition and the results can be integrated into an automatic image indexing pipeline.

Acknowledgment

This work was funded by the BMBF (German Federal Ministry of Education and Research), project INBEKI (13N10787) and Perspecting (01 IW 08002).

References

[1] K. Jung, K. I. Kim, and A. K. Jain, "Text information extraction in images and video: a survey," *Pattern Recognition*, vol. 37, no. 5, pp. 977–997, 2004.

[2] P. Dubey, "Edge based text detection for multi-purpose application," in *Intl. Conf. on Signal Processing*, Vienna, December 2006, pp. 16–20.

[3] D. Chen, J.-M. Odobez, and H. Bourlard, "Text detection and recognition in images and video frames," *Pattern Recognition*, vol. 37, no. 3, pp. 595–608, 2004.

[4] D. Karatzas and A. Antonacopoulos, "Text extraction from web images based on a split-and-merge segmentation method using colour perception," in *Int. Conf. on Pattern Recognition*, Cambridge, UK, August 2004, pp. 634–637.

[5] X. Wang, L. Huang, and C. Liu, "A video text location method based on background classification," *Int. J. on Document Analysis and Recognition*, vol. 13, pp. 173–186, 2010.

[6] J. Gllavata, R. Ewerth, and B. Freisleben, "Text detection in images based on unsupervised classification of high-frequency wavelet coefficients," in *Int. Conf. on Pattern Recognition*, Cambridge, UK, August 2004, pp. 425–428.

[7] C. Liu, C. Wang, and R. Dai, "Text detection in images based on unsupervised classification of edge-based features," in *Int. Conf. on Document Analysis and Recognition*, Seoul, Korea, August 2005, pp. 610–614.

[8] Y. Pan, X. Hou, and C. Liu, "A hybrid approach to detect and localize texts in natural scene images," *IEEE Trans. on Image Processing*, vol. 20, no. 3, pp. 800–813, March 2011.

[9] F. Bao, A. Li, and Q. Zheng, "Photo time-stamp recognition based on particle swarm optimization," in *Int. Conf. on Web Intelligence*. Beijing, China: IEEE Computer Society, September 2004, pp. 529–532.

[10] A. Li, "Fast photo time-stamp recognition based on SGNN," in *Int. Symp. on Neural Networks*, Chengdu, China, May 2006, pp. 316–321.

[11] X. Chen and H.-J. Zhang, "Photo time-stamp detection and recognition," in *Int. Conf. on Document Analysis and Recognition*, Edinburgh, Scotland, August 2003, pp. 319–322.

[12] A. Torralba, M. S. Castelhano, A. Oliva, and J. M. Henderson, "Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search," *Psychological Review*, vol. 113, no. 4, pp. 766–786, 2006.

[13] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, November 1998.

[14] E. Simoncelli and W. Freeman, "The steerable pyramid: a flexible architecture for multi-scale derivative computation," in *Int. Conf. on Image Processing*, 1995, pp. 444–447.

[15] D. S. Bloomberg, "Color quantization using octrees."

[16] C. Wolf and J.-M. Jolion, "Object count/area graphs for the evaluation of object detection and segmentation algorithms," *Int. J. Doc. Anal. Recognit.*, vol. 8, pp. 280–296, August 2006.

[17] H. Goto, "An overview of the WEOCR system and a survey of its use," in *Proc. of Image and Vision Computing*, Hamilton, New Zealand, December, pp. 121–125.