Exemplar Driven Character Recognition in the Wild

Karthik Sheshadri
sheshadri@cmu.edu
Santosh K. Divvala
santosh@ri.cmu.edu

The Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania
USA

Abstract

Character recognition in natural scenes continues to represent a formidable challenge in computer vision. Beyond variation in font, there exist difficulties in occlusion, background clutter, binarisation, and arbitrary skew. Recent advances have leveraged state of the art methods from generic object recognition to address some of these challenges. In this paper, we extend the focus to Indic script languages (e.g., Kannada) that contain large character sets (order of 1000 classes unlike 62 in English) with very low inter character variation. We identify this scenario as a fine grained visual categorization task, and present a simple exemplar based multi layered classification approach to the problem. The proposed approach beats the existing state of the art on the chars74k and ICDAR datasets by over 10% for English, and around 24% for Kannada.

1 Introduction

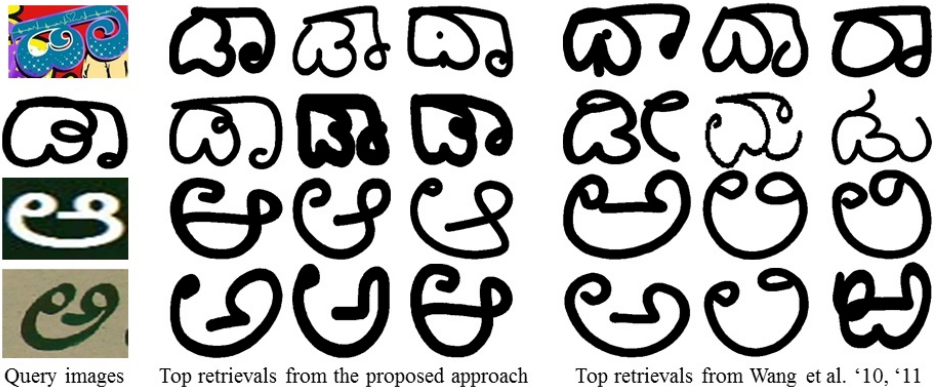
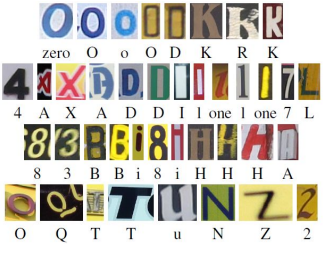


Figure 1: The problem of fine grained character recognition in unconstrained visual scenes is addressed in this paper. Note the following: (i) our approach retrieves better results than Wang *et al*. [7, 8] (ii) The query images in rows 1 and 2 are of the same character but look very different. This motivates an exemplar approach to classification. (iii) The query images in rows 3 and 4 represent different characters but look very similar, emphasizing the need for fine grained categorization approaches.

The pervasive nature of text in almost all human environments represents a powerful source of information for robots and autonomous agents to leverage in performing a variety of real world tasks. The growing ubiquity of mobile imaging devices has led to applications in various environments, for example, autonomous navigation in urban settings, vision based indoor assistance for mobile robots [13], and so on. In particular, the proliferation of online databases has led to the necessity to recognize and obscure text in street level imagery for privacy protection [14].

Scene text recognition is a challenging research problem which has only recently gained interest from the vision community. The problem is significantly more difficult than reading text from scanned images; text localization is in itself a significant challenge, and characters are often occluded and deformed, making segmentation difficult. Scene text recognition essentially constitutes four sub-problems: (1) full image text detection, (2) full image word recognition, (3) cropped word recognition, and (4) cropped character classification [12]. The first attempt at addressing the end to end scene text recognition problem was made in [7], and poor performance at the cropped character recognition level was identified as one of the primary factors contributing to overall system error.



(a) Sample English instances from chars74k

(b) Sample Kannada instances from chars74k

Figure 2: Scene text recognition is more difficult than printed text: beyond variation in font, there exists occlusion, background clutter, difficulties in segmentation and binarisation, bad resolution, arbitrary skew, etc.

Traditional optical character recognition (OCR) methods fail to perform well on characters from scene text owing to a variety of difficulties in background clutter, binarisation, and arbitrary skew (fig. 2). Recent advances [1, 7, 8] have sought to leverage generic object recognition approaches to address these challenges, but have met with limited success. For instance, [7] reports an accuracy of 58.5% at the character recognition level on the English component of the benchmark chars74k dataset.

Further, English characters group into only 62 classes [A-Z,a-z,0-9] whereas many of the world's languages (such as Chinese, Russian, Korean, etc) have several hundred classes. In particular, most Indic script languages such as Kannada exhibit large intra class diversity, while the only difference between two classes may be in a minor contour above or below the character (fig. 2).

These considerations motivate an exemplar approach to classification; one which does not seek intra class commonality among extreme examples which are essentially sub classes of their own. An exemplar approach also has the benefit of explicit correspondence: we are able not only to identify a character's class, but also infer a variety of characteristics: what font is the character? Is it printed or a sketch?, and so on. The contribution of this paper is twofold: first, an exemplar SVM based classification approach is presented which achieves accuracies over 10% higher than the state of the art on the English components of the benchmark chars74k and ICDAR-CH datasets(see fig. 3 for qualitative results). Then, a layered

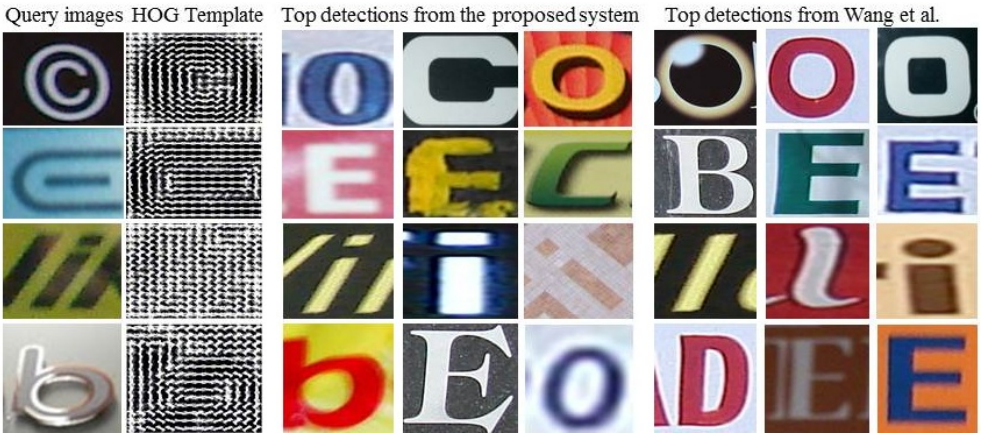


Figure 3: A qualitative comparison of the proposed exemplar SVM based system with Wang et al. [7, 8] on query images from the challenging chars74k dataset.

classification based approach is presented to illustrate how fine grained character recognition may be achieved on a generic world language such as Kannada [19]. Results are presented on the Kannada component of the chars74k dataset which demonstrate an improvement of 24% on the existing state of the art. Having demonstrated enhanced performance on both a conventional and a complex language, we argue that exemplar SVM's are a viable tool for generic character recognition, unconstrained by script or setting.

2 Related Work

Computer vision research is swiftly moving from a laboratory experimental setting to more real world, unconstrained applications. In particular, the problem of face recognition in the wild has been addressed fairly successfully [2]. Recent work also considers the pros and cons of the widely prevalent database performance benchmark evaluation of algorithms [3].

A wealth of recent advances have been made in generic object recognition, ranging from improvements in features (shape context matching [4], Histograms of Oriented Gradients [5]) to variants in the classifier. Notable is the KNN-SVM approach, which uses k nearest neighbors as an initial screening mechanism before employing SVMs for more fine grained recognition. Recent work by Ramanan *et al*. [6] presents a general taxonomy of distance metrics and shows how most distance functions in use may be approximated by a general geodesic distance. In [18], Divvala *et al*. introduced the notion of visual subcategories, as partitioning instances belonging to a basic-level category into smaller groups using discriminative feature-space clustering.

T. E. de Campos *et al*. [1] describes the first attempt at character recognition in natural scenes. They introduced the chars74k dataset of characters for natural scene images, and demonstrated that commercial OCR engines did not perform well on the new task. They also proposed a Multiple Kernel Learning based method which achieved an accuracy of 55% for English. The performance on Kannada characters however was uniformly poor. The object recognition based approach motivated by [1] was taken forward by Wang *et al*. [7],

who employed HOG features together with a Nearest Neighbor classifier. This improved over the performance in [1] by about 2%, achieving an accuracy of 57% on the 15 instance per training class benchmark of chars74k. Wang *et al*. also proposed a Bayesian inference based method [8] which did not achieve any improvement on chars74k, but improved over the approach in [7] by 12% on ICDAR-CH. While [9] reports a performance of 72% on chars74k, it does not state the number of instances used per training class. Also, [9] did not use the benchmark splits introduced and advocated by [1], making it difficult to compare results.

3 Approach

The choice and configuration of SVM's used for a particular classification problem are motivated by a number of factors: the number of classes that need to be separated, their similarity, the presence of heuristics or insights into a particular problem (see sec 3.2.2), and most importantly the feature representation.

At one extreme, it is conceivable that we could have a 62 class SVM for English, crowd all the allowed positives and negatives into this one multi-class classifier, and hope that the feature representation used is discriminative enough to define meaningful decision boundaries between them. The advantage of such an approach would be that only a single SVM needs to be run, and no decision calibration or consensus process is required. However, it is not difficult to see from previous results(see Table 1) that the features available today would not yield good results with such a classifier.

Exemplar SVM's lie at the other end of the spectrum, making individual decisions extremely easy for each classifier, and relying on subsequent decision calibration to arrive at a systemic consensus. Most current SVM based OCR systems [1] explore intermediate points of the spectrum, i.e, break the classification problem into a smaller number of classes and employ a layered classification approach using category SVM's. When one considers the intra class diversity in visual scene images, however, it is difficult even to group different positives within the same class [15, 16, 17]. This indicates that a potential way to correctly recognize these extreme examples within a class is to have a separate classifier trained for each of them, one that does not seek to establish commonality between positives which are essentially sub classes of their own.

3.1 Exemplar SVM's: A brief review

Exemplar SVM's have been recently introduced in [10] in the object recognition context. For the benefit of the reader, we include a brief review of the concepts involved and then explain details specific to our approach. The essence of the exemplar approach is that rather than seeking to establish commonality within classes, a separate classifier is learnt for each exemplar in the dataset. To make individual classification simple, linear SVM's are used and each classifier is hence an exemplar specific weight vector. Each exemplar in the dataset is resized to standard dimensions, and thence HOG features are densely extracted to create a rigid template x_E. A set of negative samples N_E are created by the same process from classes not corresponding to the exemplar.

Each classifier (w_E,b_E) maximizes the separation between x_E and every window in N_E.

This is equivalent to optimizing the convex objective[10]:

$$\Omega_E(w,b) = \| w \|^2 + C_1 h(w^T x_E + b) + C_2 \sum_{x \in N_E} h(-w^T x - b), \quad (1)$$

where $h(.)$ indicates the hinge loss function, and C_1, C_2 are constants.

3.2 Calibrating Exemplar SVM's for Character Recognition

In return for simpler classification at the level of each exemplar, we must now deal with the problem of decision calibration: combining decisions from independently trained and hence non compatible classifiers. In this work, we explore the following two calibration methods.

3.2.1 Calibration based on SVM scores

In the spirit of [10], we adopt an "on the fly" calibration method, using positives selected by each exemplar based on SVM scores. Exemplars which achieve low scores on ground truth labelled query images from the validation set are suppressed by moving the decision boundary in their requisite classifier towards the exemplar and well performing exemplars are boosted by moving the decision boundary in their classifier away from the exemplar. Given a detection 'x' and the learned sigmoid parameters α_E,β_E, the calibrated detection score for each exemplar E is as follows: $f(x|\alpha_E,\beta_E,w_E) = \frac{1}{1+e^{-\alpha_E\left(w_E^T x - \beta_E\right)}}$. This rescaling and shifting of the decision boundary conditions each classifier to fire only on visually similar examples, and underlines the explicit correspondence offered by the exemplar SVM based approach. A caveat of this approach is that a large training database is required, because the system makes decisions based on largest score.

3.2.2 Calibration based on affine motion estimation

This calibration approach is based on a simple observation: variations in font and shape essentially constitute small affine transformations. Characters from visual scenes are often affine warps of characters from normal text: they are oriented differently, different character contours are irregularly shaped, and are of different sizes, etc. Hence on thinned character images, one could compute affine motion [11] between train and test characters, and minimize the sum of absolute differences to refine candidate choices obtained by simple max voting of the exemplar SVM's. Our proposed approach is summarized as follows: (i) count the number of positive votes in favour of each class, computed based on a preselected threshold[1] (ii) extract the top k of these classes, and perform affine motion estimation $M_{E_C,Q}$ between the thinned binarized query image Q and every exemplar E_C, C being the class of the exemplar, in the training subset corresponding to the top k classes (iii) recognize the character as that class which minimizes the sum of absolute differences (SAD) between the test character and any exemplar in the training subset corresponding to top k classes. Equation (2) illustrates the approach:

$$B = \arg\min_{C \in C_K} \{E_C - M_{E_C,Q} Q\} \quad (2)$$

[1] Since for the purposes of max voting we are only interested in obtaining a binary yes/no from each classifier, this threshold is 0. Note that this choice of threshold is relatively consistent across classifiers and hence avoids the chicken and egg problem of choosing a threshold across non compatible SVM's.

where B is the computed belief class of query image Q, and $M_{E_C,Q}$ is the affine transformation matrix which warps Q with respect to E_C.

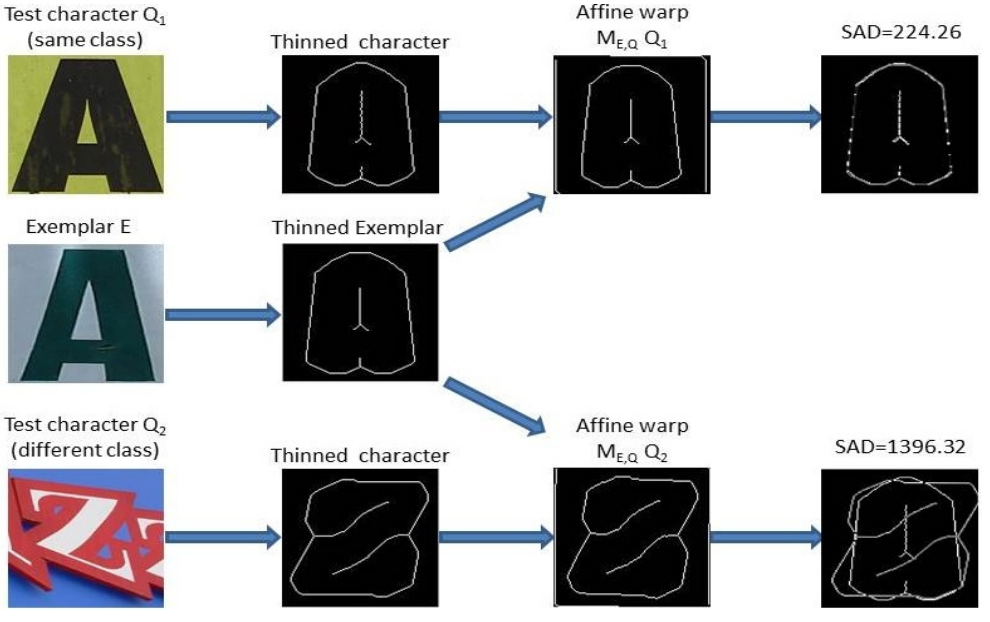


Figure 4: The Affine calibration process. Lucas-Kanade tracking is employed to estimate affine warps between thinned test characters and exemplars. Low SAD indicates a good match.

3.3 Layered approach to Indic Language Classification

Kannada is an Indic script language in which modifier glyphs from the 16 vowels are used to alter the 34 base consonants, creating a total of 578 characters. In addition, a consonant emphasis glyph exists for each consonant. This results in nearly 650 classes. Advantage is taken of the fact that many of these are similar, motivating a layered approach. First, we run the test character against an SVM system of 34 classes, corresponding to each of the consonants. For each of these SVM's, the negatives are taken from the other 33 classes, and include vowel/ consonant emphasis modified images. The output of this initial system is the base consonant class of the test character. We then address the problem of identifying the modifier glyphs, by running SVM's in which the negatives are taken from the *same* base class, but from *different* sub classes corresponding to the *same* base class.

4 Results

We evaluate our approach on the challenging chars74k and ICDAR-CH datasets. The chars74k dataset consists of character images harvested from natural scenes, hand written text, and computer synthesized data, corresponding to a total of 74 thousand characters. It was introduced in [1] and is the de facto benchmark for unconstrained character classification. ICDAR-CH is a robust character classification dataset [12] and remains a challenging benchmark for the evaluation of character recognition algorithms. Previous experimental benchmarks[1]

Table 1: Our results (English) on chars74k and ICDAR-CH, and comparison to baseline methods.

Model	Chars74k-5	Chars74k-15	ICDAR-CH
HOG+ESVM+AFF	**48.43 ± 2.40**	**69.66**	**70.53**
HOG+ESVM+on fly calib	27.76 ± 1.74	60.00	66.67
HOG+NN+AFF	47.61 ±0.81	64.22	63.59
HOG+ESVM	16.33 ± 2.33	44.68	41.44
HOG+NN[7]	45.33 ± 0.99	57.50	52
NATIVE+FERNS[8]	−−	54	64
MKL[1]	−−	55.26	−−

on chars74k advocate the use of 5 and 15 training examples per class, and specify a series of train and test splits. This is considerably to the disadvantage of the proposed system, since a score calibrated exemplar system needs a large training database to identify visually similar examples. Note also that while [1, 7, 8] can use their entire training set, the proposed method needs to split the allowed training data in order to create a validation set for on fly calibration, and these examples cannot hence be used. On the chars74k-5 benchmark, we use 3 examples for training and 2 for validation, while we use 10 examples for training and 5 for validation on the chars74k-15 benchmark.

In order to fairly compare the proposed approach with the methods in use, we present separate results using calibration "on the fly" with a large training set, and also results with the chars74k-5, 74k-15 and ICDAR-CH benchmarks using both the affine motion based calibration scheme (sec 3.2.2), and also on fly calibration (sec 3.2.1).

Table 1 shows results from the proposed method alongside the current state of the art on both the ICDAR-CH dataset and the English subset of the chars74k dataset. In the case of chars74k-5, there is not much training data available and hence the calibration of SVM's based on scores is imperfect. However, this approach still retains a flavor of the nearest neighbor method used in [7], and achieves an accuracy of 28%. There is often equal confidence in two or more classes, and the subsequent selection of specious but false candidates causes max voting of the SVM's to break down. The accuracy of the max voting method on this benchmark is 16%. However, advantage is taken of the fact that the test character's true label is still probably within the top k candidates. Affine calibration plays a key role in chars74k-5, and yields an accuracy of 48%, 3% better than [7].

However, the chars74k-15 benchmark allows for much more flexibility in the training data. A sample vote distribution for test letter 'H' is shown in Fig. 5(b). It is easily inferred from Fig. 5(b) that the confidence in the proposed candidates increases with the size of the training set. Without calibration, however, max voting suffers from the same problem as the category SVM: intra class diversity is too large to separate classes in the feature space. Max voting hence yields a similar accuracy to the MKL method described in [1]. On the fly calibration using SVM scores helps restore the "nearest neighbor++" flavor of the exemplar SVM, exploiting the larger training set and surpassing the state of the art by about 3%. This improved performance notwithstanding, the benchmark of 15 instances per class is still too small for the genuine explicit correspondence characteristic of on the fly calibration to make itself felt. We hence turn once again to the affine calibration approach, which yields an accuracy of nearly 70%, 13% above the state of the art. Results on the ICDAR-CH dataset

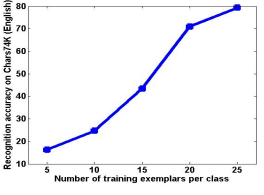
Table 2: Results (Kannada) on chars74k.

Feature	Train on Hnd, Test on Hnd	Train on Hnd, Test on Img
HOG+ESVM+AFF	54.13	1.76
SC[1]	29.88	3.49

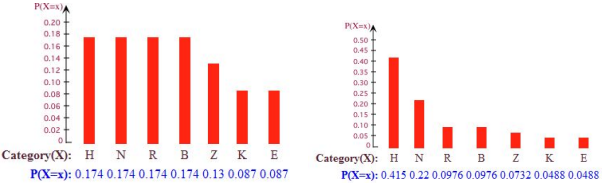
follow a similar pattern to that of chars74k-15.

In table 2, the performance of the proposed approach is compared with the results on the Kannada component of chars74k reported in [1] (see Fig. 6 for qualitative results). Reference [1] advocates the use of shape context matching for Kannada, and introduces two separate hand-written and scene text dataset components, Hnd and Img. Reference [1] trains on Hnd, and tests on Hnd and Img. [2]

Given a test set with a sufficient degree of similarity to the training data, the exemplar SVM approach is essentially a “nearest neighbor ++", and this is apparent from the results on Hnd. The proposed method is over 10% above the SC approach used in [1]. However, the paradigm of trying to find visually similar examples breaks down completely when the test data has almost no representation in the training set, as is the case with column 2 of Table 2. Neither approach performs well on this benchmark.



(a) Recognition accuracy with on-the-fly calibration



(b) Vote distribution for character ‘H’ on chars74k-5 (left) & chars74k-15 (right)

Figure 5: (a) Performance increases with the size of the allowed training set. (b) Confidence in the selected candidates increases with training data

The above results compared the proposed approach alongside the state of the art on standard benchmarks which did not allow enough training data to demonstrate the power of exemplar SVM’s in finding visually similar correspondences. Figure 5(a) illustrates the performance of the ESVM+ on the fly calibration approach with a dataset size ranging from 5 to 25 instances per class. Correspondences are selected based on SVM scores, and improve as the size of the training data set. Note that for the purposes of this demonstration, we did not utilize the train and test splits specified in [1], but rather selected visually dissimilar training classes. Note also that although the performance on chars74k-5 and chars74k-15 are below that of [7], accuracy increases to a peak of 79% given 25 exemplars per training class(see fig. 6 for qualitative results).

[2]Given the complexity of the Kannada language, the motivation for this design choice is not immediately clear, however the benchmark is accepted in this work for the sake of comparison.

Figure 6: Query images from the benchmark chars74k dataset, and their top 6 retrievals from our system. An incorrect retrieval is shown in the bottom row.

5 Conclusion

This paper explored the utility of a fine grained categorization motivated exemplar SVM approach to unconstrained character recognition. Two separate methods for decision calibration were presented, with applicability according to the size of the allowed training set. Results were presented on the English and Kannada components of Chars74k, which rise significantly higher than the state of the art beyond a threshold in training size. Motivated by the performance on two languages ranging from conventional to extremely complex, we argue that leveraging fine grained categorization and generic object recognition approaches is a promising research direction for character recognition unconstrained by language or setting.

References

[1] T. de Campos, B. Babu, and M. Varma. Character recognition in natural images. In: VISAPP 2009.

[2] http://vis-www.cs.umass.edu/lfw/

[3] A. Torralba, and A. Efros. An unbiased look at Database Bias. In: CVPR 2011.

[4] S. Belongie, J. Malik, and J. Puzicha. Shape Matching and Object Recognition Using Shape Contexts. PAMI (2002) Vol. 24 (24): pp 509-521.

[5] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection, In: CVPR 2005.

[6] D. Ramanan and S. Baker. Local Distance Functions: A Taxonomy, New Algorithms, and an Evaluation. PAMI (2011) Vol. 33 (4): pp 794-806.

[7] K. Wang and S. Belongie. Word Spotting in the Wild. In: ECCV 2010.

[8] Kai Wang, Boris Babenko, and Serge Belongie. End to End Scene Text Recognition. In: ICCV 2011.

[9] L. Neumann and J. Matas. A Method for Text Localization and Recognition in Real World Images. In: ACCV 2010.

[10] Tomasz Malisiewicz, Abhinav Gupta, and Alexei A. Efros. Ensemble of Exemplar-SVMs for Object Detection and Beyond. In: ICCV 2011.

[11] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. Image Understanding Workshop (1981) pp 121-131.

[12] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young. ICDAR 2003 robust reading competitions. In: ICDAR 2003.

[13] M. Trabelsi, N. Ait-Oufroukh, and S. Lelandais. Localization Method for a Rehabilitation Mobile Robot using Visual and Ultrasonic Information. In: ICORR 2007.

[14] Andrea Frome, German Cheung, Ahmad Abdulkader, Marco Zennaro, Bo Wu, Alessandro Bissacco, Hartwig Adam, Hartmut Neven, and Luc Vincent. Large-scale privacy protection in Google Street View. In: ICCV 2009.

[15] Lubomir Bourdev and Jitendra Malik. Poselets: Body Part Detectors Trained Using 3D Human Pose Annotations. In: ICCV 2009.

[16] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L.Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR 2009.

[17] O, Chum and A. Zisserman. An Exemplar Model for Learning Object Classes. In: CVPR 2007.

[18] Santosh K. Divvala, Alexei A. Efros and Martial Hebert. How important are 'Deformable Parts' in the Deformable Parts Model?. arXiv:1206.3714. 2012.

[19] http://en.wikipedia.org/wiki/Kannada