Application of Selective Search to Pose estimation

Ujwal Krothapalli
Department of Electrical and
Computer Engineering
Virginia Tech
Blacksburg, Virginia 24061
ujjwal@vt.edu

Abstract

This paper will apply the generic objectness measure to sample windows from a given image and then feed them through a human pose estimation pipeline. The Objectness step will provide windows in the BUFFY and PARSE datasets and some of them will contain people. This approach would be better than a scanning window technique in terms of computational speed as the filter convolution step of the pose estimation algorithm can be computed in selective regions of the input image. We discuss the results obtained using the above described pipeline and another baseline over the BUFFY and PARSE datasets.

1. Introduction

Cascade based detectors are quite efficient in detecting specific objects. By rejecting regions of the image that are not as relevant scanning window techniques can benefit in terms of computation time. In this paper we discuss a method to use 'Objectness' to speed up the human pose estimation problem. The two main contributions by this paper are,

1) Improvement in speed by 30 for pose detection (including objectness computation).

2) New objectness score is 40 percent faster.

1.1. Objectness

Almost all current day object detectors employ a scanning window detector. The authors [1] propose a generic object detector. This approach uses the intuition that the object and its background differ in their visual cues. A closed boundary usually encloses an object, this intuition is used as a part of a multitude of visual cues to come up with a metric which the authors call 'Objectness'.

1.2. Pose estimation

Detecting the pose of a rigid object is a problem that can be solved by using part templates.The parts based method uses multiple templates for representing various regions within the object boundary. Pose estimation of an articulated object (human) is however a difficult task. A fully working pose estimation tool can be used to automate many tasks like, monitoring people driving cars and machinery. Surveillance in general can also benefit a lot from such a tool. One of the popular ways to solve this problem is by using individual templates for different parts of the body and having geometric constraints on the pairs of templates [2]. The dominant methods for pose estimation have involved parametrization of the parts based on pixel locations and their orientations. Approaches based on parts parametrized purely based on their locations also have achieved good results in the object detection area [3], [4]. This paper will use the implementation of [5].

2. Pipeline

This paper will use the existing implementation of the objectness code and generate a new measure of objectness which is faster to compute. And the pose estimation algorithm has also been described in this section.

2.1. Objectness computation

Previous work in this area focused on using interest point detectors, which are helpful in recognizing specific points in a given image. Salient features are class dependent and do not constitute a generic object detector. The authors [6]have used a more advanced approach to find a single generic object in a given image. However this approach would fail in a multi-object detection framework. The authors in [1] introduced a new visual cue called superpixel straddling, which helps detect the object boundaries better. And they improve upon the original saliency implementation of [7]. The authors [1] use multiple visual cues to

train and detect objects using their objectness measure using a bayesian framework. The authors also speed up the class dependent object detectors by replacing the scanning window approach and detecting objects in the windows returned by their algorithm. This greatly reduces the computation time especially when convolving with filters as is the case with both part based and template based methods. The visual cues used in the paper are,

1)Multi-scale saliency: This approach extends the previous work by [7] to the multi scale level to be able to detect objects at multiple scales.

2) Color Contrast: Changes in contrast usually help understand how different the objects are from their surroundings. This measure was computed in the LASB color space.

3) Edge Density: The measure used the density of the edges at the borders of the object windows. A simple canny edge detector was used.

4) Superpixel Straddling: One of the most common ways to identify closed boundary objects is through superpixels . The authors use the intuition that closed boundary characteristic of objects rests on the superpixels.

The training stage is accomplished by using the positive windows and the negative windows the cues are extracted and the max likelihood function is formulated and as the visual cues are complementary a joint likelihood using naive bayes is accomplished. About 50 images containing bounding boxes for various objects are used in training the objectness detector.

This paper uses only two of the above visual cues. The multi-scale saliency for 5 scales and a new edge detection algorithm described in [8].

2.2. Pose estimation

Estimating the pose of a human is a hard computer vision problem. Especially because the size and shape of the hind and fore limbs vary from person to person. Variance is also introduced because of the clothing, camera viewpoint changes and foreshortening. To capture all these variations a complex model needs to be built. Also the amount of data that has to be learnt from is vast. The approximation used by [5] models the variation by approximating the effects of foreshortening using a mixture of pictorial structures that do not encode orientation. The authors use a dynamic programming approach to share the computation across mixtures at the time of inference. By capturing the effects of global geometry on the local parts and partial occlusion of the parts their model performs better than most baselines.

Part based models have been used for rigid body recognition. As there is no movement among the different parts of the object, a global mixture model can capture the variation in different views [4]. The approach taken by [5] is suitable for non-rigid object detection. By using mixtures for the local part templates they generate an exponential number

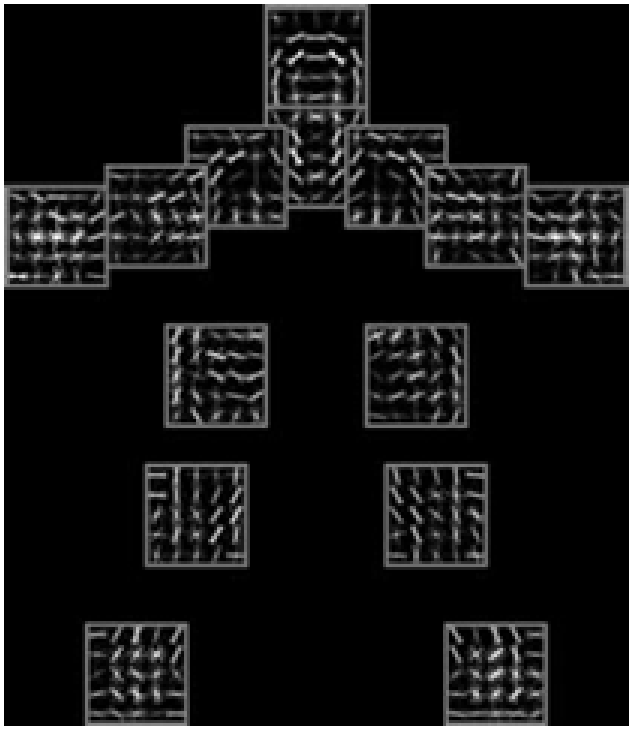

Figure 1. The HOG templates visualized for a 14 part human model from the implementation of [5]

of global mixtures. A prior for co-occurrence of these local parts is computed to accomodate for the 'likeliness' of a local part co-occurring with another(usually the neighbor). Supervised learning was employed to learn the structured model. The structure or grammar (elbow is connected to the forearm, which is connected to the wrist etc.) of the model is captured at learning time.

One of the standard ways to evaluate the efficacy of pose estimation algorithms is by computing the percentage of correctly localized parts(PCP) first introduced by [9]. Numerous other implementation exist because of the vague specifications introduced by the authors also the implementation assumed that people were detected beforehand. The authors of [5] developed a more 'robust' evaluation method called APK. We will be using only the 'gold standard' APK in this paper.The authors in [5] use a tree structure to encode the spatial structure and train their model discriminatively using positive and negative sets of images.

2.2.1 Learning

The authors in [5] rely on a joint learning framework to improve the accuracy of detection. Intuitively the makes sense and detection only one weak template at a time will result in a poor performance compared to summing up the scores for multiple such templates.

2.2.2 The objective function

To capture the various deformations and viewpoint changes an approximation can be made by using a mixture model. For an image I let the pixel location of part i and the corresponding mixture component (t_i) be $l_i = (x, y)$. The mixture component is essentially a rotated version of the template. For a K number of mixtures we have $t = t_1, t_2....t_k$. The scoring function would also have to capture the co-occurrence relationship for all the parts.

$$S(t) = \sum_{i \in V} b_i^{t_i} + \sum_{ij \in E} b_{ij}^{t_i, t_j} \quad (1)$$

Where $b_i^{t_i}$ is a prior for the mixture model for part i and the co-occurrence is captured by $b_{ij}^{t_i,t_j}$, the value would be high for mixtures that are 'consistent' and low for any inconsistent orientations. Rigidity between the rigid parts of the body is also encoded through this co-occurrence computation. To compute the score for various local part templates and location the following equation is used.

$$S(I, l, t) = S(t) + \sum_{i \in V} w_i^{t_i}.\phi(I, l_i) + \sum_{ij \in E} w_{ij}^{t_i,t_j}.\psi(l_i - l_j) \quad (2)$$

2.2.3 Inference

This is a maximization problem where $S(I, l, t)$ has to be maximized over locations and mixtures. Because a tree model has been used, the computation can be carried out efficiently by using dynamic programming. Let $z_i = (l_i, t_i)$

$$S(I, z) = S(t) + \sum_{i \in V} \phi_i(I, z_i) + \sum_{ij \in E} \psi_{ij}(z_i - j_j) \quad (3)$$

$$\phi_i(I, z_i) = w_i^{t_i}.\phi(I, l_i) + b_i^{t_i} \quad (4)$$

$$\psi_{ij}(z_i - j_j) = w_{ij}^{t_i,t_j}.\psi(l_i - l_j) + b_{ij}^{t_i,t_j} \quad (5)$$

This model resembles a Markov random field. The maximum can be computed using dynamic programming. This approach will result in multiple detections in a single image. By using a non-maximal suppression strategy the 'best' detection can be obtained.

3. Datasets

For evaluation purposes the authors have used the Image Parse dataset [10] and the Buffy Stickman dataset [9], [11]. Both the datasets have been hand annotated, each part in the people in the images has been labeled. The authors [5] achieve state of the art results on these datasets.

4. Experiments

The objectness code by [1] takes about 2.8 seconds to return 10 windows containing objects for a standard image. The new objectness code using the structured edge prediction and 5 scale saliency takes about 1.6 seconds for the same. The average precision or APK evaluation of the BUFFY and PARSE datasets was 77 and 64 percent without the objectness pipeline. With the objectness pipeline the time to process an image was about 2.5 seconds compared to 3.5 seconds without the objectness for BUFFY and about 3.8 seconds from 4.8 seconds for the PARSE dataset. The APK after using the objectness pipeline was around 67 for BUFFY and 51 for PARSE. This was after various window sampling techniques have been applied like, aspect ratio of the windows and their relative size.

Figure 2 is showing some of the qualitative examples of the complete pipeline.

A second baseline using [4] was used to provide a bounding box across the people detected and then the pose estimation algorithm was run. The APK for BUFFY was 75 and PARSE was 62. The evaluation metric only scores one person in the image, and BUFFY and PARSE have multiple people, so when a mistake is made by the person detector (it detected the person that was not annotated) the accuracy suffered. The speed was about the same as the original implementation of the pose estimation algorithm. As the DPM person detector took about 2.5 seconds per image. The HOG computation was not shared between the implementations, however this would have made only a minor difference in speed.

5. Future work

Future work will explore the use of diverse sampling methods for windows and this might result in a better sampling of windows.

Images can be downsampled to measure objectness, this will speed things up and will not affect the computation severely as multi scale saliency is being computed and the speed would improve considerably.

Class specific detectors can also be build using the objectness approach, intuitively this makes sense as all the training images for the objectness algorithm can be replaced by class specific images.

References

[1] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?" in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 73–80.

[2] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *International Jour-*

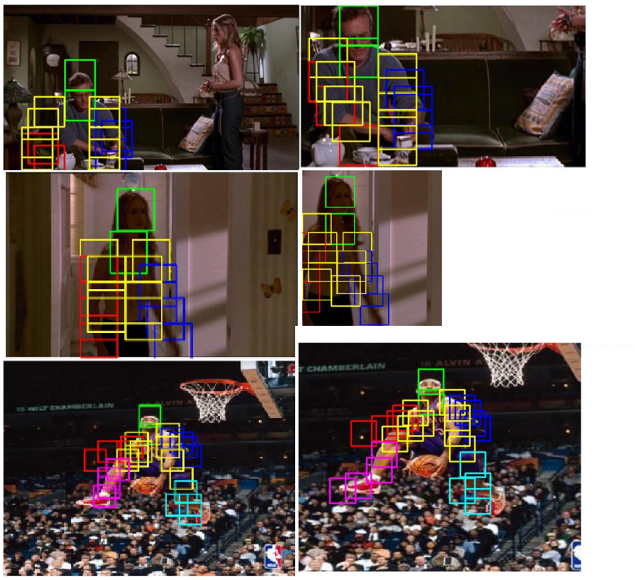

Figure 2. Results: Pose estimated for people in images and the objectness sampled windows. Left column contains the actual images and Right column contains the ones sampled by the objectness measure

nal of Computer Vision, vol. 61, no. 1, pp. 55–79, 2005.

[3] L. Bourdev and J. Malik, "Poselets: Body part detectors trained using 3d human pose annotations," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1365–1372.

[4] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[5] Y. Yang and D. Ramanan, "Articulated human detection with flexible mixtures-of-parts," 2012.

[6] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 2, pp. 353–367, 2011.

[7] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8.

[8] P. Dollár and C. L. Zitnick, "Structured forests for fast edge detection," 2013.

[9] V. Ferrari, M. Marin-Jimenez, and A. Zisserman, "Progressive search space reduction for human pose estimation," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[10] D. Ramanan, "Learning to parse images of articulated bodies," *Advances in Neural Information Processing Systems*, vol. 19, p. 1129, 2007.

[11] M. Eichner, M. Marin-Jimenez, A. Zisserman, and V. Ferrari, "2d articulated human pose estimation and retrieval in (almost) unconstrained still images," *International journal of computer vision*, vol. 99, no. 2, pp. 190–214, 2012.