Nested Pictorial Structures

Steve Gu and Ying Zheng and Carlo Tomasi

Department of Computer Science,
Duke University, NC, USA 27705
{steve,yuanqi,tomasi}@cs.duke.edu

Abstract. We propose a theoretical construct coined nested pictorial structure to represent an object by parts that are recursively nested. Three innovative ideas are proposed: First, the nested pictorial structure finds a part configuration that is allowed to be deformed in geometric arrangement, while being confined to be topologically nested. Second, we define nested features which lend themselves to better, more detailed accounting of pixel data cost and describe occlusion in a principled way. Third, we develop the concept of constrained distance transform, a variation of the generalized distance transform, to guarantee the topological nesting relations and to further enforce that parts have no overlap with each other. We show that matching an optimal nested pictorial structure of K parts on an image of N pixels takes $O(NK)$ time using dynamic programming and constrained distance transform. In our MATLAB/C++ implementation, it takes less than 0.1 seconds to do the global optimal matching when $K = 10$ and $N = 400 \times 400$. We demonstrate the usefulness of nested pictorial structures in the matching of objects of nested patterns, objects in occlusion, and objects that live in a context.

1 Introduction

We study explicit, visual nesting relations in images. Nested visual patterns are commonly seen in our daily life. To name a few examples: a pupil is contained in an eye contained in a face contained in a person. A portrait is contained in a painting contained in a frame contained in a wall. In general, a *texture* is often contained in a *part* contained in an *object* contained in a *context* contained in a *scene*. In the literature of computer vision, most nesting relations are modeled implicitly. In this paper we present nested pictorial structure to *explicitly* model the nesting relation among textures, parts, objects, and contexts. Figure 1 illustrates that a nested pictorial structure represents parts in a possibly complex topological nesting relation.

1.1 Three New Ideas

To the best of our knowledge, the nesting relation is rarely explicitly modeled in the computer vision literature, and in particular within the context of a pictorial structure. This paper introduces a mathematical and computational model that

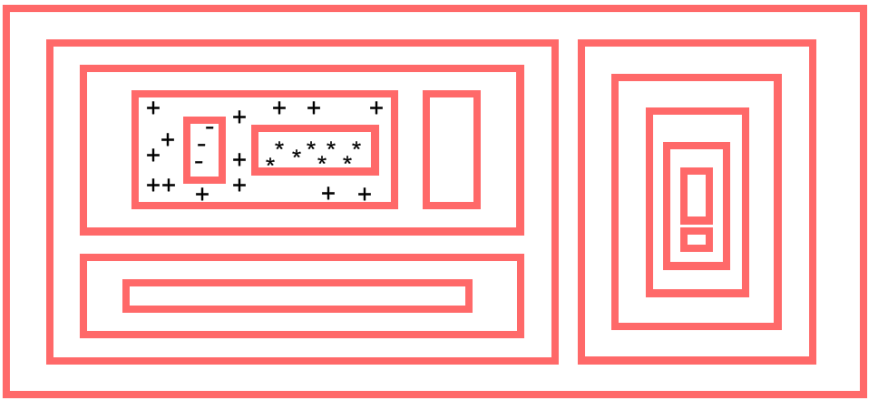

Fig. 1. Each red rectangle represents a part that can be recursively nested. The part arrangement is allowed to deform significantly in the image domain while keeping the nesting relations unchanged. Each part has its own cost function (e.g. $+, -, *$, etc), and a single sliding window without nesting would over-count the data cost. This issue is addressed by nested features in a principled way.

accounts for the *recursive* nesting relations among parts and objects. Three innovative ideas are presented:

First, we define *nested pictorial structure*, a concept that is rooted in the conventional pictorial structure (e.g. [1, 2]), but enforces that parts conform to a given topological nesting relation. Moreover, parts are allowed to deform in their geometric arrangement and the matching of a nested pictorial structure is cast as a global optimization on the pixel level. This enriches the common understanding of the pictorial structure framework and suggests possibly new directions and novel computational schemes for the optimal matching.

Second, we define the notion of *nested features*. Nested features provide a better, more detailed accounting of pixel costs when parts are recursively nested. For instance, if part B is contained in part A, the spatial extent of A then needs to exclude the spatial extent of B. The conventional way of using a sliding window for the cost evaluation may involve significant "over-counting" of pixel costs when one part resides within another part (Figure 1 and Figure 2). Consequently, a typical detection algorithm may fail to localize an object under significant occlusion due to the lack of features and to the fact that the visible spatial extent is distributed on the boundaries rather than the interior. Nested features address this problem in an explicit and principled way.

Third, we develop non-overlapping constraints and the *constrained distance transform*, a variation of the generalized distance transform by Felzenszwalb and Huttenlocher [3], for the efficient, global matching of a nested pictorial structure. The constrained distance transform guarantees that no two parts overlap with each other in the deformable template matching. This transform enhances the

stability in the geometric part arrangement and rules out the otherwise possible degenerate cases in which parts are collapsed together. Our algorithm runs in $O(NK)$ time for matching a nested pictorial of K parts on an image of N pixels. This is asymptotically optimal and enables real time computation.

1.2 Related Work

Pictorial structures, a.k.a. deformable models or constellation models, have been studied for decades, spanning from early work [1, 4–7] to more recent contributions in recognition [8–14] and scene understanding [15]. A pictorial structure is typically represented as a graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ where $\mathcal{V}$ is a set of symbolic parts. Each part $v \in \mathcal{V}$ is further represented by a rectangular window $W_v \subset \mathbb{Z}^2$ of fixed size in the 2D image domain. Let $f_v(W_v)$ be the cost of placing part v at window W_v. Let the deformation cost between part u and v be $\delta(W_u, W_v)$ that measures the discrepancy between the vector linking u to v and their template configuration. The commonly used objective is to find an optimal part allocation through the following global optimization:

$$\{\hat{W}_v\} = \arg \min_{\{W_v\}_{v \in \mathcal{V}}} \left\{ \underbrace{\sum_{v \in V} f_v(W_v)}_{\text{data cost}} + \lambda \underbrace{\sum_{(u,v) \in \mathcal{E}} \delta(W_u, W_v)}_{\text{deformation cost}} \right\} \tag{1}$$

where λ is a regularization parameter that balances the data cost and deformation cost. The intuition is clear: we want an arrangement of parts that matches a given object template in both appearance (minimizes the data cost) and spatial configuration (minimizes the deformation cost).

The computational challenge is to efficiently compute an optimal matching of a pictorial structure against a template. For instances, in some previous work [6, 9], the location of each part is confined to a sparse set of points for practical processing. Felzenszwalb and Huttenlocher [16, 2] show that when the spatial constraints form a tree and when the deformation cost δ is in a special form, the optimal allocation of a pictorial structure takes linear time in the size of an image, multiplied by the number of parts. Moreover, the optimization works on the image pixel level. The underlying technique of [2] is coined the generalized distance transform [3], and is equivalent to computing the lower envelope of an array of cones or parabolas. Combined with HoG features [17] and latent SVM, Felzenszwalb et al. [13] demonstrate a state-of-the-art recognition system [18].

Also remotely related are the works in occlusion reasoning [19–21] and context or scene recognition [22–24]. The nested pictorial structure is expected to be applicable but not limited to these two areas. The notable difference is that instead of detecting occlusion boundaries, the occlusion itself is inherently encoded in the nested model and is therefore enforced in an explicit way. Instead of dividing an image into a spatial grid of cells, nested models represent an image as a foliation of windows, one enclosing another, arranged in a possibly complex

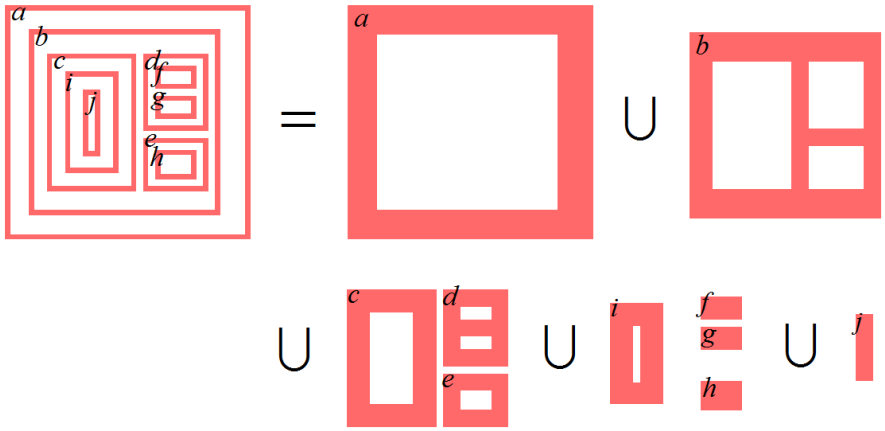

Fig. 2. A nested pictorial structure conformal to a nesting tree. Part a is maximal because it is not contained in any other part. Shaded connected regions are the spatial extent of each part.

inclusion relation. Instead of being rigid, nested parts are allowed to deform significantly in their geometric arrangement. The concept of a nested pictorial structure is new to the best of our knowledge.

2 Nested Pictorial Structures

We use $a \prec b$ to represent that part a is *enforced* to be contained in part b. That is, any window W_a associated to part a must be contained in any window W_b associated to part b: $W_a \subseteq W_b$. We have the following definition:

Definition 1 (Nesting Graph) *Given a set of parts $\mathcal{V}$ and their inclusion relations $\prec$, the nesting graph is represented by $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ where $\mathcal{E}$ is the edge set. Edge $(u, v) \in \mathcal{E}$ if $u \prec v$ and there does not exist $l \in \mathcal{V}$ such that $u \prec l \prec v$. Part u is called maximal if u is not contained in any other part. A window set $\{W_v\}_{v \in \mathcal{V}}$ is said to be conformal to $\mathcal{G}$ if $W_u \subseteq W_v$ for all $(u, v) \in \mathcal{E}$.*

It follows that the constructed graph $\mathcal{G}$ is a forest (a collection of trees) by observing that the graph is acyclic and each child node has at most one parent node by construction. If there is only a single maximal part, $\mathcal{G}$ is reduced to a tree. Figure 2 visualizes each level of a nesting tree. In the figure each connected component represents a single node. The nesting graph encodes topological nesting relations and complements the pairwise spatial relations defined in conventional pictorial structures.

2.1 Nested Features

We describe the visual content of each part using its histogram representation. Applicable descriptors include color histogram, histogram of oriented gradient (HoG) [17], their possible concatenations and more. The difference is that the spatial extent of each part is inherently nested and the histogram needs to be modified accordingly. Let $h : \Omega \to \mathbb{Z}^L$ be a histogram of L bins calculated in the image region Ω. $h[i]$ is the number of occurrences of i in Ω and is nonnegative. Histograms have an important property that for any $A, B \subseteq \Omega$:

$$h(A \cup B) = h(A) + h(B) - h(A \cap B) \tag{2}$$

A natural consequence is that for $A \subseteq B$,

$$h(B \setminus A) = h(B) - h(A) \tag{3}$$

The subtractive property of Equation (3) is important for computing the histogram of each part in the nested model. Let $C_v \triangleq \{W_u \mid (u, v) \in \mathcal{E}\}$ be the set of child windows of part v. We require that no two child windows overlap with each other, that is, $\bigcap S = \emptyset$ for any $S \subseteq C_v$. This constraint is imposed in the global matching of a pictorial structure (See Section 2.4 for details). The histogram of part v is therefore computed as:

$$h_v(W_v; C_v) \triangleq h(W_v) - \sum_{W \in C_v} h(W) \tag{4}$$

We normalize h_v to $\tilde{h}_v$ by dividing the area of part v:

$$\tilde{h}_v(W_v; C_v) = \frac{h_v(W_v; C_v)}{|W_v \setminus \bigcup C_v|} = \frac{h(W_v) - \sum_{W \in C_v} h(W)}{|W_v| - \sum_{W \in C_v} |W|} \tag{5}$$

so that $\sum_{l=1}^{L} \tilde{h}_v[l] = 1$. $\tilde{h}_v$ may be interpreted as a sample probability distribution (e.g. color distribution) of part v.

2.2 Data Cost

We score each nested part using a linear SVM classifier. Let α_v and β_v be the trained normal vector of the learnt classification boundary and the bias term respectively for part v. We define the data cost of part v as:

$$\varphi_v(W_v; C_v) = \left\langle \alpha_v, \tilde{h}_v(W_v; C_v) \right\rangle + \beta_v \tag{6}$$

Let $\lambda_v = |W_v| - \sum_{W \in C_v} |W|$ be the area of part v. We can further expand Equation (6) by substituting Equation (5):

$$\begin{aligned} \varphi_v(W_v; C_v) &= \frac{1}{\lambda_v} \left\langle \alpha_v, h(W_v) - \sum_{W \in C_v} h(W) \right\rangle + \beta_v \\ &= \frac{1}{\lambda_v} \langle \alpha_v, h(W_v) \rangle - \sum_{W \in C_v} \frac{1}{\lambda_v} \langle \alpha_v, h(W) \rangle + \beta_v \end{aligned} \tag{7}$$

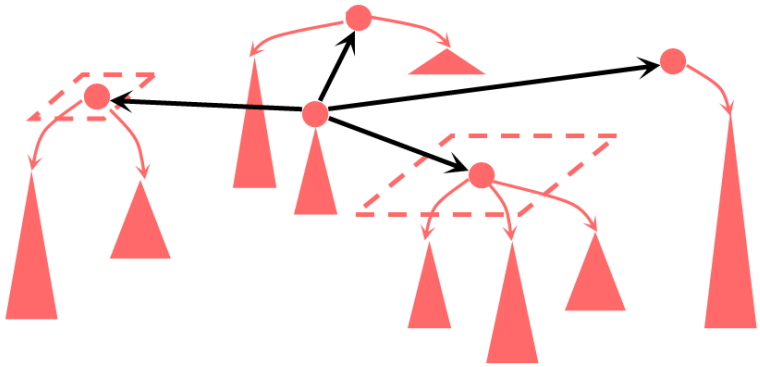

Fig. 3. Additional spatial constraints combine the nesting forest (red) and any spanning tree of the maximal parts (black).

The term $\frac{1}{\lambda_v}\langle \alpha_v, h(W)\rangle$ can be expanded as:

$$\frac{1}{\lambda_v}\langle \alpha_v, h(W)\rangle = \frac{1}{\lambda_v}\sum_{l=1}^{L}\alpha_v[l]h[i] = \sum_{p\in W}\frac{\alpha_v[l[p]]}{\lambda_v} \tag{8}$$

where $l[p]$ is the bin index of pixel p. This way of converting vector dot product to per-pixel computation is also presented in Lampert et al. [25]. We define:

$$\xi_v(p) = \frac{\alpha_v[l[p]]}{\lambda_v} \tag{9}$$

Then, Equation (7) can be rewritten as:

$$\varphi_v(W_v; C_v) = \sum_{p\in W_v}\xi_v(p) - \sum_{W\in C_v}\sum_{p\in W}\xi_v(p) + \beta_v \tag{10}$$

The total data cost of the window set $\{W_v\}_{v\in\mathcal{V}}$ is:

$$\begin{aligned}\varphi\left(\{W_v\}_{v\in\mathcal{V}}\right) &= \sum_{v\in\mathcal{V}}\varphi_v(W_v; C_v)\\ &= \sum_{v\in\mathcal{V}}\left[\underbrace{\sum_{p\in W_v}\xi_v(p)}_{f_v(W_v)} - \sum_{W\in C_v}\underbrace{\sum_{p\in W}\xi_v(p)}_{f_v(W)} + \beta_v\right]\\ &= \sum_{v\in\mathcal{V}}\left[f_v(W_v) - \sum_{W\in C_v} f_v(W)\right] + \text{const.}\end{aligned} \tag{11}$$

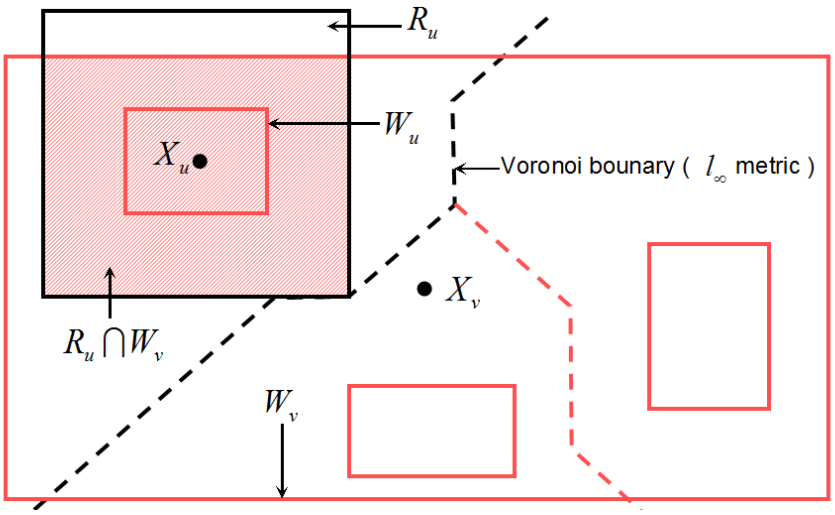

Fig. 4. The shaded region is the intersection of W_v and R_u. W_u is allowed to drift within the shaded region only. Note that the region does not assume a fixed position. It moves with the parent window W_v during deformable template matching.

where $f_v(W)$ is the summation of all the pixel costs ξ_v within W. We remark that $f_v(W)$ can be evaluated in $O(1)$ time by the use of an integral image representation. The constant numbers can be safely removed because they have no effect on the global minimization.

2.3 Deformation Cost

We also model the typical arrangement of a nested pictorial structure. Similar to [13] we connect nested parts by conceptual springs in a tree configuration. In fact, the nesting forest can be directly used for specifying the tree configuration. More spatial constraints are allowed if there are multiple maximal parts. Let $\mathcal{E}'$ be any spanning tree (e.g. minimal spanning tree) of the set of maximal parts. Then, the edge set $\mathcal{E}' \cup \mathcal{E}$ remains a directed tree (Figure 3). Any maximal part can serve as a root node. The deformation cost of a set of windows $\{W_v\}_{v \in \mathcal{V}}$ is:

$$\delta(\{W_v\}_{v \in \mathcal{V}}) = \sum_{(u,v) \in \mathcal{E} \cup \mathcal{E}'} \sum_{(u,v) \in \mathcal{E} \cup \mathcal{E}'} \underbrace{| \, \|(X_u - X_v) - (T_u - T_v)\|_1 \, |}_{\delta(W_u, W_v)} \qquad (12)$$

where X_u and X_v are the 2D centroid of W_u and W_v. $T_u - T_v$ is the given template vector linking part u to part v. In the definition we use the L_1 norm although other metrics such as squared Euclidean [3] or its truncation apply equally well and enjoy the same computational complexity.

2.4 Non-Overlapping Constraints

It is important to constrain child parts so as not to overlap, in order to maintain the validity of Equation (4) and to avoid the degeneracies of parts collapsing into a single region. This non-overlapping constraint is also new in the context of pictorial structures. To quantify the range each part is allowed to deform without overlapping, we compute the distance transform spanned by parts' windows under the l_∞ norm in the template image[1]. The rectangle (centered at X_u) that first touches Voronoi boundaries within the Voronoi region of X_u thus bounds the maximal free space of W_u. Let this rectangle be R_u. The non-overlapping constraint can be expressed as $W_u \subseteq R_u$. Note that R_u and W_u share the same centroid by construction. Moreover, since the nesting relation is $W_u \subset W_v$, we can combine both nesting and non-overlapping constraints into a single constraint: $W_u \subseteq R_u \cap W_v$. See Figure 4 for illustration.

2.5 Global Optimization

The inputs to the optimization are: a nesting forest $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, the template vectors $\{T_u\}_{u \in \mathcal{V}}$, the cost functions $\{f_u\}_{u \in \mathcal{V}}$, the spanning tree among maximal parts $\mathcal{E}'$, the range constraints $\{R_u\}_{u \in \mathcal{V}}$ and a regularization parameter λ. The optimal allocation of a nested pictorial structure is:

$$\min_{W_u \subseteq R_u \cap W_v \text{ for each } (u,v) \in \mathcal{E}} \left\{ \sum_{u \in \mathcal{V}} \left[f_u(W_u) - \sum_{W \in C_u} f_u(W) \right] + \lambda \sum_{(u,v) \in \mathcal{E} \cup \mathcal{E}'} \underbrace{\|(X_u - X_v) - (T_u - T_v)\|_1}_{\delta(W_u, W_v)} \right\} \tag{13}$$

The nested pictorial structure is reduced to the conventional one when $f_v = f_u$ for each $(u, v) \in \mathcal{E}$ and when no nesting or overlapping constraints are imposed. In general, the nested model extends the conventional pictorial structure by ensuring that parts conform to a nesting forest and child parts do not overlap in their geometric arrangement.

3 Dynamic Programming

We show that the objective in Equation (13) can be computed in $O(NK)$ time using dynamic programming and a technique called constrained distance transform. Here N is the image size and K is the number of nested parts. Let $O_u(W_u)$ be the optimal cost when part u is placed at window W_u. Then, for each parent

[1] In MATLAB, this is easily achieved by the command bwdist(I,'chessboard'). The distance transform is linear in the image size.

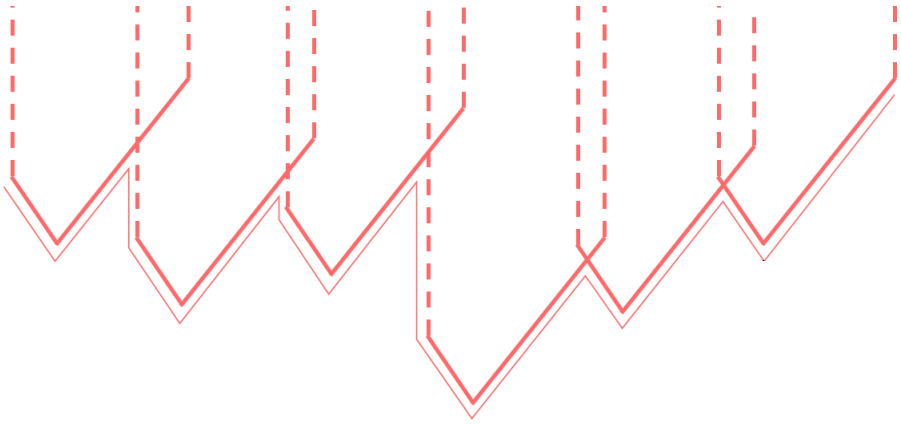

Fig. 5. The lower envelope of cones with sides beyond a range lifted to infinity.

node p we derive the following recursive state equation:

$$\Delta_p^c(W_p) = \min_{W_c \subseteq W_p \cap R_c} \{O_c(W_c) - f_p(W_c) + \lambda\delta(W_c, W_p)\} \tag{14}$$

$$F_p^c(W_p) = \min_{W_c} \left\{ O_c(W_c) + \lambda \underbrace{\|(X_p - X_c) - (T_p - T_c)\|}_{\delta(W_c, W_p)} \right\} \tag{15}$$

$$O_p(W_p) = f_p(W_p) + \sum_{c:(c,p)\in\mathcal{E}} \Delta_p^c(W_p) + \sum_{c:(c,p)\in\mathcal{E}'} F_p^c(W_p) \tag{16}$$

Equation (15) can be evaluated in $O(N)$ thanks to the generalized distance transform [3]. Equation (14) can be solved in $O(N)$ time too. Let $(x_c, y_c), (x_p, y_p)$ be the centroid of W_c and W_p. The constraint $W_c \subseteq R_c \cap W_p$ can be conveniently expressed by $B \leq x_c - x_p \leq A$ and $C \leq y_c - y_p \leq D$ where A, B, C, D are constants determined by the size of W_c, R_c and W_p (Figure 4). Equation (14) is equivalent to the following 2D constrained distance transform:

$$g(x, y) = \min_{A \leq x' - x \leq B, C \leq y' - y \leq D} \{f(x', y') + \lambda\,[|x' - x - E| + |y - y' - F|]\} \tag{17}$$

where E and F are constants. Geometrically the transform is equivalent to the lower envelope of cones with asymmetric sides. The 1D version is shown in Figure 5. Similar to the generalized distance transform of Felzenszwalb and Huttonlocher [3], it is easy to show that 2D constrained distance transform can be computed in $O(N)$ time as well. Therefore, the optimal matching of a nested pictorial structure of K parts takes $O(KN)$ time.

4 Proof of Concept

We demonstrate the usefulness of the proposed nested pictorial structure in localizing highly complex patterns, objects being occluded, and objects in a context. We use a linear SVM as the underlying classifier for training and the testing is similar to what is used in the recognition system of Felzenszwalb et al.[13]. Sample results from the 12 categories of [26] are shown in Figure 6. Some natural applications are: Finding a scene where A is in front of B or Finding an object of particular nested patterns. For instance, we choose the example of matching a surfer against the water, and a clown fish nested in a sea anemone. Our method is also expected to be useful for matching animal or human faces where the nested features are able to separate features associated to eyes, nose, mouth, and features associated to the skin and hair, and other fine scene characteristics. In our MATLAB/MEX implementation it takes less than 0.1 seconds to do the matching in a 400×400 image when the number of parts is less than 10. The running time grows linearly with the number of image pixels and the number of object parts, both asymptotically and experimentally.

5 Conclusions

Nested pictorial structure finds a part arrangement that is flexible in spatial configuration, but confined to be nested accordingly. Moreover, we define nested features which lend themselves to better accounting of pixel data cost, where a sliding window technique may likely over-count. We develop an $O(NK)$ algorithm for the optimal matching of a nested pictorial structure of K parts on an image of N pixels.

Promising but anecdotal results are shown. Many questions remain open. For instance, it is not clear how to automatically construct a reasonable nesting relation from training data. The current nesting relation has to be specified manually in the training data and is therefore difficult for large scale annotation. A deeper study of matching nested patterns is expected to be useful for applications in general recognition, tracking, and instance based query systems.

Acknowledgement: This work is supported by the Army Research Office under Grant No. W911NF-10-1-0387 and by the National Science Foundation under Grant IIS-10-17017.

References

1. Fischler, M., Elschlager, R.: The representation and matching of pictorial structures. **22** (1973) 67–92
2. Felzenszwalb, P., Huttenlocher, D.: Pictorial structures for object recognition. IJCV **61** (2005) 55–79
3. Felzenszwalb, P., Huttenlocher, D.: Distance transforms of sampled functions. Technical Report TR2004-1963, Cornell Computing and Information Science (2004)

4. Yuille, A., Hallinan, P., Cohen, D.: Feature extraction from faces using deformable templates. IJCV **8** (1992) 99–111
5. Burl, M., Weber, M., Perona, P.: A probabilistic approach to object recognition using local photometry and global geometry. In: ECCV. (1998) 628–641
6. Weber, M., Welling, M., Perona, P.: Towards automatic discovery of object categories. In: CVPR. (2000) 2101–2108
7. Coughlan, J., Yuille, A., English, C., Snow, D.: Efficient deformable template detection and localization without user initialization. Computer Vision and Image Understanding **78** (2000) 303–319
8. Cootes, T., Edwards, G., Taylor, C.: Active appearance models. IEEE PAMI **23** (2001) 681–685
9. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: IEEE CVPR. (2003) 264–271
10. Crandall, D., Felzenszwalb, P., Huttenlocher, D.: Spatial priors for part-based recognition using statistical models. In: IEEE CVPR. (2005) 10–17
11. Amit, Y., Trouvé, A.: Pop: Patchwork of parts models for object recognition. IJCV **75** (2007) 267–282
12. Leibe, B., Leonardis, A., Schiele, B.: Robust object detection with interleaved categorization and segmentation. IJCV **77** (2008) 259–289
13. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE PAMI **32** (2010) 1627–1645
14. Zhu, L., Chen, Y., Yuille, A., Freeman, W.: Latent hierarchical structural learning for object detection. In: IEEE CVPR. (2010) 1062–1069
15. Pandey, M., Lazebnik, S.: Scene recognition and weakly supervised object localization with deformable part-based models. In: ICCV. (2011)
16. Felzenszwalb, P., Huttenlocher, D.: Efficient matching of pictorial structures. In: IEEE CVPR. (2000) 2066–2073
17. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE CVPR. (2005) 886–893
18. Everingham, M., Gool, L.V., Williams, C., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. (http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html)
19. Gao, T., Packer, B., Koller, D.: A segmentation-aware object detection model with occlusion handling. In: IEEE CVPR. (2011) 1361–1368
20. Sundberg, P., Brox, T., Maire, M., Arbelaez, P., Malik, J.: Occlusion boundary detection and figure/ground assignment from optical flow. In: IEEE CVPR. (2011) 2233–2240
21. Humayun, A., Aodha, O., Brostow, G.: Learning to find occlusion regions. In: IEEE CVPR. (2011) 2161–2168
22. Woodcock, C., Harward, V.: Nested-hierarchical scene models and image segmentation. Internal Journal of Remote Sensing **13** (1992)
23. Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. IJCV **42** (2001) 145–175
24. Li, L., Socher, R., Fei-Fei, L.: Towards total scene understanding: Classification, annotation and segmentation in an automatic framework. In: IEEE CVPR. (2009) 2036–2043
25. Lampert, C., Blaschko, M., Hofmann, T.: Efficient subwindow search: A branch and bound framework for object localization. IEEE PAMI **31** (2009) 2129–2142
26. Griffin, G., Holub, A., Perona, P.: Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology (2007)

Fig. 6. Example matching of nested pictorial structures. We select 12 categories of images from Caltech256, see [26]. Each category is divided into separate sets for training and testing. In the training step, we ask a user to mark a number of nested rectangular windows on each training image, and a linear SVM classifier is trained for each category from both positive and negative samples using nested feature descriptors. The feature descriptor of each part is a concatenation of histogram of gradients and colors in that part, excluding histograms associated to any child parts. In the matching step, the normal vector derived from the trained linear SVM classifier is used to obtain the per-pixel cost of Equation (9). The matching is performed on images from the test set for each category. Similar to the recognition framework of Felzenszwalb et al. [13], we build an image pyramid of 10 levels ranging from twice the size of the original image to half of the original image size. Matching a nested pictorial structure is then performed on each derived image and the one that receives the minimal cost is selected and reshaped back. In some images the rectangles are attached to a single object. In others the rectangles include multiple objects and their surrounding context (e.g. clown fish and anemone, the Sphinx and pyramid, etc).