Integrating Spatial Layout of Object Parts into Classification without Pairwise Terms

Application to Fast Body Parts Estimation from Depth Images

Mingyuan Jiu, Christian Wolf and Atilla Baskurt
Université de Lyon, CNRS, INSA-Lyon, LIRIS, UMR5205, 69621, Villeurbanne, France

Keywords: Object Detection, Pose Estimation, Spatial Layout, Unary Terms, Randomized Decision Forest, Kinect.

Abstract: Object recognition or human pose estimation methods often resort to a decomposition into a collection of parts. This local representation has significant advantages, especially in case of occlusions and when the "object" is non-rigid. Detection and recognition requires modeling the appearance of the different object parts as well as their spatial layout. The latter can be complex and requires the minimization of complex energy functions, which is prohibitive in most real world applications and therefore often omitted. However, ignoring the spatial layout puts all the burden on the classifier, whose only available information is local appearance. We propose a new method to integrate the spatial layout into the parts classification without costly pairwise terms. We present an application to body parts classification for human pose estimation. As a second contribution, we introduce edge features from gray images as a complement to the well known depth features used for body parts classification from Kinect data.

1 INTRODUCTION

Object recognition is one of the fundamental problems in computer vision, as well as related problems like face detection and recognition, person detection, and associated pose estimation. Local representations as collections of descriptors extracted from local image patches are very popular. This representation allows robustness against occlusions and permits non-rigid matching of articulated objects, like humans and animals.

For object recognition tasks, the known methods in the literature vary in their degree of usage of spatial relationships, between methods not using them at all, for instance the bags of visual words model (Sivic and Zisserman, 2003), and rigid matching methods using all available information, e.g. based on RANSAC (Fischler and Bolles, 1981). The former suffer from low discriminative power, whereas the latter only work for rigid transformations and cannot be used for articulated objects. Methods for non-rigid matching exist. Graph-matching and hyper-graph matching, for instance, restricts the verification of spatial constraints to neighbors in the graph. However, non trivial formulations require minimizing a complex energy functions and are NP-complete (Torresani et al., 2008; Duchenne et al., 2009).

Pictorial structures, deformable parts based models, have been introduced as early as in 1973 (Fischler and Elschlager, 1973). The more recent seminal work creates a Bayesian parts based model of the object and its parts, where the possible relative parts locations are modeled as a tree structured Markov random field(Felzenszwalb and Huttenlocher, 2005). The absence of cycles makes minimization of the underlying energy function relatively fast — of course much slower than a model without pairwise terms. In (Felzenszwalb et al., 2010) the Bayesian model is replaced with a more powerful discriminative model, where scale and relative position of each part are treated as latent variables and searched by Latent SVM.

A similar problem occurs in tasks where joint object recognition and segmentation is required. Layout CRFs and extensions model the object as a collection of local parts (patches or even individual pixels), which are related through an energy function (Winn and Shotton, 2006). However, unlike pictorial structures, the energy function here contains cycles which makes minimization more complex, for instance through graph cuts techniques. Furthermore, the large number of labels makes the expansion move algorithms inefficient (Kolmogorov and Zabih, 2004). In the original paper (Winn and Shotton, 2006), and

Jiu M., Wolf C. and Baskurt A..
Integrating Spatial Layout of Object Parts into Classification without Pairwise Terms - Application to Fast Body Parts Estimation from Depth Images.
DOI: 10.5220/0004278206260631
In *Proceedings of the International Conference on Computer Vision Theory and Applications* (VISAPP-2013), pages 626-631
ISBN: 978-989-8565-47-1

as in our proposed work, the unary terms are based on randomized decision forests. Another related application which could benefit from this contribution is full scene labelling (Farabet et al., 2012).

Pose estimation methods are also often naturally solved through a decomposition into body parts. A preliminary pixel classification step segments the object into body parts, from which the joint positions can be estimated in a second step. The well known method used for the MS Kinect system completely ignores the spatial relationships between the objects parts and puts all the classification burden on the pixel wise working classifier (Shotton et al., 2011). The decision function to be learned by the classifier is complex and therefore requires a learning machine with a complex architecture, which is difficult to learn. The good performance of the system has been obtained with an extremely large training set of $2 \cdot 10^9$ training vectors extracted from 1 million images and training on a computation cluster with 1000 nodes.

In this paper we propose a method which segments an object into parts through pixelwise classification and which integrates the spatial layout of the part labels. Like the methods ignoring the spatial layout, it is extremely fast as no additional step needs to be added to pixelwise classification and no energy minimization is necessary. The (slight) additional computational load only concerns learning at an offline stage. The goal is not to compete with methods based on energy minimization, which is impossible through pixelwise classification only. The objective is to improve the performance of pixelwise classification by using all available information during learning.

Classical learning machines working on data embedded in a vector space, like neural networks, SVM, randomized decision trees, Adaboost etc., are in principal capable of learning arbitrary complex decision functions, if the underyling prediction model (architecture) is complex enough. In reality the available amount of training data and computational complexity limit the complexity which can be learned. In most cases only few data are available with respect to the complexity of the problem. It is therefore often useful to impose some structure on the model. In this work we propose to use prior knowledge in the form of the spatial layout of the labels to add structure to the decision function learned by the learning machine.

The contributions of this paper are twofold:

- The integration of the spatial layout of part labels into learning machines, in particular randomized decision forests;
- We introduce features extracted from edges calculated on the gray image and show that they can provide valuable complementary information to the traditional depth features.

The paper is organized as follows: section 2 presents the learning procedure which integrates the spatial layout into a randomized decision forest. Section 3 introduces edge comparison features which can complement the classical depth features for pose estimation. Section 4 explains the experiments on pose estimation, and section 5 finally concludes.

2 LEARNING OBJECT PART CLASSIFIERS FROM SPATIAL LAYOUTS

We consider problems where the pixels i of an image are classified as belonging to one of L target labels by a learning machine whose alphabet is $\mathcal{L} = \{1 \dots L\}$. To this end, descriptors F_i are extracted on each pixel i and a local path around it, and the learning machine takes a decision $l_i \in \mathcal{L}$ for each pixel. A powerful prior can be defined over the set of possible labellings for a spatial object. Beyond the classical Potts model known from image restoration (Geman and Geman, 1984), which favors equal labels for neighboring pixels over unequal labels, additional (soft) constraints can be imposed. Labels of neighboring pixels can be supposed to be equal, or at least compatible, i.e. belonging to parts which are neighbors in the spatial layout of the object. In computer vision this kind of constraints is often modeled soft through the energy potentials of a global energy function:

$$E(l_1, \dots, l_N) = \sum_i U(l_i, F_i) + \mu \sum_{i \sim j} D(l_i, l_j) \quad (1)$$

where the unary terms $U(\cdot)$ integrate confidence of a pixelwise employed learning machine and the pairwise terms $D(\cdot,\cdot)$ are over couples of neighbors $i \sim j$. In the case of certain simple models like the Potts model, the energy function is submodular and the exact solution can be calculated in polynomial time using graph cuts (Kolmogorov and Zabih, 2004). Taking the spatial layout of the object parts into account results in non-submodular energy functions which are difficult to solve. Let's note that even the complexity of the submodular problem (quadratic on the number of pixels in the worst case) is far beyond the complexity of pixelwise classification with unary terms only.

The goal of our work is to improve the learning machine in the case where it is the only source of information, i.e. no pairwise terms are used for classification. Traditional learning algorithm in this context are supervised and use as only input the training feature vectors f_i as well as the training labels l_i, where

i is over the pixels of the training set. We propose to provide the learning machine with additional information, namely the spatial layout of the labels of the alphabet $\mathcal{L}$.

2.1 Randomized Decision Forests

In this paper we focus on randomized decision forests (RDF) as learning machines, because they have shown to outperform other learning machines in this kind of problem and they have become very popular in computer vision lately (Shotton et al., 2011). Decision trees, as simple tree structured classifiers with decision and terminal nodes, suffer from overfitting. Randomized forests, on the other hand, overcome this drawback by integrating distributions over several trees.

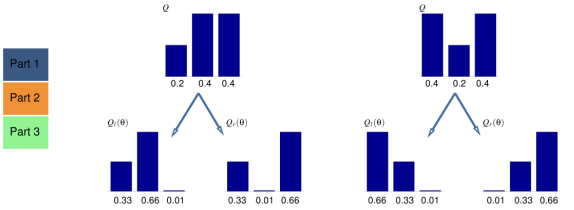
The classical learning algorithm for RDFs (Lepetit et al., 2004) is training each tree separately, layer by layer. Each layer is also trained separately, which allows the training of deep trees with a complex prediction model. The drawback of this approach is the absence of any gradient on the error during training. Instead, training maximizes the gain in information based on Shannon entropy. In the following we give a short description of the classical training procedure.

We describe the version of the learning algorithm from (Shotton et al., 2011) which jointly learns features and the parameters of the tree, i.e. the thresholds for each decision node. We denote by θ the set of all learned parameters (features and thresholds) for each decision node. For each tree, a subset of training instances is randomly sampled with replacement.

1. Randomly sample a set of candidates θ.
2. Partition the set of input vectors into two sets, one for the left child and one for the right child according to the threshold $\tau \in \theta$. Denote by Q the label distribution of the parent and by $Q_l(\theta)$ and $Q_r(\theta)$ the label distributions of the left and the right child node, respectively.
3. Choose θ with the largest gain in information:

$$\begin{aligned}\theta^* &= \arg\max_{\theta} \quad G(\theta) \\ &= \arg\max_{\theta} \quad H(Q) - \sum_{s\in\{l,r\}} \frac{|Q_s(\theta)|}{|Q|} H(Q_s(\theta))\end{aligned} \tag{2}$$

   where $H(Q)$ is the Shannon entropy from class distribution of set Q.
4. Recurse the left and right child until the predefined level or largest gain is arrived.



	Classical: $G(\theta)$=0.37	Classical: $G(\theta)$=0.37
	Spatial: $G'(\theta)$=0.22	Spatial: $G'(\theta)$=0.31
(a)	(b)	(c)

Figure 1: An example of three parts: (a) part layout; (b) a parent distribution and its two child distributions for a given θ; (c) a second more favorable case. The entropy gain for the spatial learning cases are given with $\lambda = 0.3$.

2.2 Spatial Learning for Randomized Decision Forests

In what follows we will integrate the additional information on the spatial layout of the object parts into the training algorithm, which will be done without using any information on the training error. Let us first recall that the target alphabet of the learning machine is $\mathcal{L} = \{1 \dots L\}$, and then imagine that we create groups of pairs of two labels, giving rise to a new alphabet $\mathcal{L}' = \{11, 12, \dots 1L, 21, 22, \dots 2L, \dots, LL\}$. Each of the new labels is a combination of two original labels. Assuming independent and identically distributed (i.i.d.) original labels, the probability of a new label ij consisting of the pair of original labels i and j is the product of the original probabilities, i.e. $p(ij) = p(i)p(j)$. The Shannon entropy of a distribution Q' over the new alphabet is therefore

$$H(Q') = \sum_k -p(k)\log p(k) \tag{3}$$

where k is over the new alphabet. This can be expressed in terms of the original distribution over the original alphabet:

$$H(Q') = \sum_{i,j} -p(i)p(j)\log[p(i)p(j)] \tag{4}$$

We can now separate the new pairwise labels into two different subsets, the set of neighboring labels $\mathcal{L}'^1$, and the set of not neighboring labels $\mathcal{L}'^2$, with $\mathcal{L}' = \mathcal{L}'^1 \cup \mathcal{L}'^2$. We suppose that each original label is neighbor of itself. In the same way, a distribution Q' over the new alphabet can be split into two different distributions Q'^1 and Q'^2 from these two subsets.

Then a learning criterion can be defined using the gain in information obtained by parameters θ as a sum over two parts of the histogram Q', each part being calculated over one subset of the labels:

$$G'(\theta) = \lambda\, G'^1(\theta) \quad + \quad (1-\lambda)\, G'^2(\theta) \tag{5}$$

where

$$G'^{i}(\theta) = H(Q'^{i}) - \sum_{s\in\{l,r\}} \frac{|Q'^{i}_{s}(\theta)|}{|Q'^{i}|} H(Q'^{i}_{s}(\theta)) \quad (6)$$

Here, λ is a weight, and $\lambda < 0.5$ in order to give separation of non neighboring labels a higher priority.

Let's consider a simple parts based model with three parts numbered from 1 to 3 shown in figure 1. We suppose that part 1 is a neighbor of 2, that 2 is a neighbor of 3, but that 1 is not a neighbor of 3. Let's also consider two cases where a set of tree parameters $\theta = \{u, v, \tau\}$ splits a label distribution Q into two distributions, the left distribution $Q_l(\theta)$ and the right one $Q_r(\theta)$.

The distributions for the two different cases are given in figure 1b and 1c, respectively. Here, we did not compare the the individual values for Shannon entropy gain between classical measure and spatial measure, as the former is calculated from the unary distribution and the latter on a pairwise distribution. However, the difference between two cases can be observed using the same measure. It can been seen that the classical measure is equal for both cases: the entropy gains are both 0.37. If we take into account the spatial layout of the different parts, we can see that the entropy gain is actually higher in the second case ($G'(\theta)$=0.31) than the first case ($G'(\theta)$=0.22) when setting λ=0.3. In the first case, the highest gain in information is obtained for parts 2 and 3, which are neighbors. However, the highest gain comes from parts 1 and 3 which are not neighbors in the second case. This is consistent with what we advocate above that a higher priority is set to non-neighbor labels.

3 DEPTH AND EDGE FEATURES

In (Shotton et al., 2011), depth features have been proposed for pose estimation from Kinect depth images. One of their main advantages is their simplicity and their computational efficiency. Briefly, at a given pixel x, the depth difference between two offsets centered at x is computed:

$$f_\theta(I, \mathbf{x}) = d_I(\mathbf{x} + \frac{\mathbf{u}}{d_I(\mathbf{x})}) - d_I(\mathbf{x} + \frac{\mathbf{v}}{d_I(\mathbf{x})}) \quad (7)$$

where $d_I(\mathbf{x})$ is the depth at pixel $\mathbf{x}$ in image I, parameters $\theta = (\mathbf{u}, \mathbf{v})$ are two offsets and are normalized by the current depth for depth-invariance. A single feature vector contains several differences, each comparison value being calculated from a different pair of offsets $\mathbf{u}$ and $\mathbf{v}$. These offsets are learned during training together with the prediction model, as described in section 2.

3.1 Edge Features

Psychophysical studies show that we can recognize a object only with its contour, In (Shotton et al., 2008), contour is defined as the outline (silhouette) together with the internal edges of the object, which enable to represent the spatial structure of the object. Here we extend this concept further by introducing edge comparison features extracted from the grayscale image. We propose two different types of features based on edges, the first using edge magnitude, and the second using edge orientation.

In our settings, we need features whose positions can be sampled by the training algorithm. However, contours are usually sparsely distributed, which means that comparison features can not directly be applied to edge images. Our solution to this problem is inspired by chamfer distance matching, which is a classical method to measure the similarity between contours (Barrow et al., 1977). We compute a distance transform on the edge image, where the value of each pixel is the distance to its nearest edge. Given a grayscale image I and its binary edge image E, the distance transform DT_E is computed as:

$$DT_E(\mathbf{x}) = \min_{\mathbf{x}':E(\mathbf{x}')=1} ||\mathbf{x} - \mathbf{x}'|| \quad (8)$$

The distance transform can be calculated in linear time using a two-pass algorithm.

The edge magnitude feature is defined as:

$$f^{EM}_\theta(\mathbf{x}) = DT_E(\mathbf{x} + \mathbf{u}) + DT_E(\mathbf{x} + \mathbf{v}) \quad (9)$$

where $\mathbf{u}$ and $\mathbf{v}$ are the same in (7). This feature indicates the existence of edges near two offsets.

Edge orientation features can be computed in a similar way. In the procedure of distance image, we can get another orientation image O_E, in which the value of each pixel is the orientation of its nearest edge:

$$O_E(\mathbf{x}) = Orientation\left(\arg\min_{\mathbf{x}':E(\mathbf{x}')=1} ||\mathbf{x} - \mathbf{x}'||\right) \quad (10)$$

The feature is computed as the difference in orientation for two offsets:

$$f^{EO}_\theta(\mathbf{x}) = O_E(\mathbf{x} + \mathbf{u}) - O_E(\mathbf{x} + \mathbf{v}) \quad (11)$$

where the minus operator takes into account the circular nature of angles. We discretize the orientation to alleviate the effect of noise.

The objective of both features is to capture the edge distribution at specific locations in the image, which will be learned by the RDF.

4 EXPERIMENTS

We performed experiments for an application of pose estimation. We would like to point out that the learning method can be applied to any parts based model which integrates pixelwise classification with random forests, for instance methods for joint object recognition and segmentation.

The proposed algorithm has been evaluated on the *CDC4CV Poselets* dataset (Holt et al., 2011). Our goal is not to beat the state of the art in pose estimation, but to show that spatial learning is able to improve pixelwise classification of parts based models. The dataset contains upper body poses taken with Kinect and consists of 345 training and 347 test depth images. The authors also supplied corresponding annotation files which contain the locations of 10 articulated parts: head(H), neck(N), left shoulder(LS), right shoulder(RS), left elbow(LE), left hand(LHA), right elbow(RE), right hand(RHA), left hip(LH), right hip(RH). We created groundtruth segmentations through nearest neighbor labeling. In our experiments, the left/right elbow (LE,RE) and hand (LHA,RHA) parts were extended to left/right upper arm (LUA,RUA) and forearm (LFA, RFA) parts, we also defined the part below the waist as other (the black area in the Figure 2b).

Unless otherwise specified, the following parameters have been used for RDF learning: 3 trees each with a depth of 9; 2000 randomly selected pixels per image, roughly distributed across the body; 4000 candidate pairs of offsets; 22 candidate thresholds; offsets and thresholds have been learned separately for each node in the forest. For spatial learning, 28 pairs of neighbors have been identified between the 10 parts based on a pose where the subject stretches his arms. The parameter λ was set to 0.4.

We evaluate our method at two levels: pixelwise classification and pairs of parts recognition. Pixelwise decisions are directly provided by the random forest. Part localizations are obtained from the pixelwise results through pixel pooling. We create a posterior probability map for each part from the results on RDF. After non-maximum suppression and low pass filtering, the location with largest response is used as an estimate of the part, and then the positions of pairs of detected parts are calculated, which approximately correspond to joints and serve as the intermediate pose indicator. In the following, we denote them by the pair of neighboring parts.

Table 1 shows classification accuracies of the three settings. A baseline has been created with classical RDF learning and depth features. Spatial learning with depth features only and with depth and edge features together are shown in table 1. We can see that that spatial learning can obtain a performance gain, although the layout is used in the prediction model and no pairwise terms have been used. Figure 2 shows some classification examples, which demonstrates that spatial learning makes the randomized forest more discriminative. The segmentation output is cleaner, especially at the borders.

Table 1: Results on body part classification in pixelwise level: D=deph features; E=edge features.

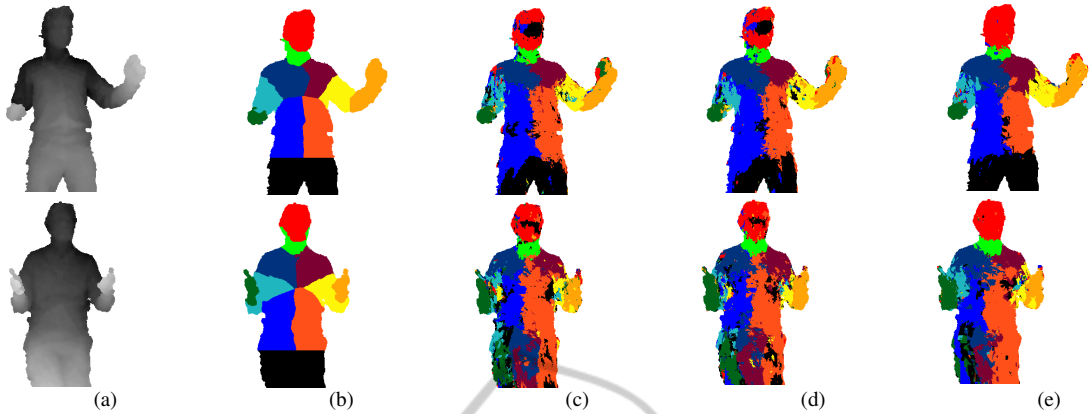
		Accuracy
Classical RDF with D		60.30%
Spatial D	$\lambda = 0.4$	61.05%
Spatial D+E	$\lambda = 0.4$	67.66%

At part level, we report our results of pairs of parts according to the estimation metric by (Ferrari et al., 2008): a pair of groundtruth parts is matched to a detected pair if and only if the endpoints of the detected pairs lie within a circle of radius r=50% of the length of the groundtruth pair and centered on it. Table 2 shows our results on part level using several settings. It demonstrates that spatial learning improves recognition performance for most of parts.

The experiments at both pixelwise and part level demonstrate that spatial learning makes randomized forest more discriminative by integrating the spatial layout into its prediction model. This proposition is very simple and fast to implement, as the standard pipeline can be still used. Only the learning method has been changed, the testing code is unchanged. There is no additional computational burden whatsoever during testing; a slight increase in computational complexity can be observed for learning. No complex discrete optimization problems need to be solved.

5 CONCLUSIONS

In this paper, we proposed a novel learning algorithm for randomized decision forests which integrates information on the spatial layout of target labels. The classification algorithm is of exactly the same computational complexity, a slightly higher computational burden is put on the learning stage. We applied our algorithm on the body part classification, although any other application requiring the segmentation of an object into parts may benefit from the contribution. Another contribution extends the well known depth comparison features to edge comparison features obtained from grayscale images. Results show that RDF indeed benefits from the integration of the spatial layout of parts and the edge features.



(a) (b) (c) (d) (e)

Figure 2: Examples of the pixelwise classification: each row is an example, each column is a kind of classification results, (a) test depth image; (b) part segmentation; (c) classical classification; (d) spatial learning with depth features; (e) spatial learning with depth features and edge features.

Table 2: Correct rate(%) on pairs of parts for different feature settings in part level: D=deph features; E=edge features.

		H-N	LS-RS	LS-LUA	LUA-LFA	RS-RUA	RUA-RFA	LS-LH	RS-RH	LH-RH	Ave.
Classical E		46.69	0.29	20.46	2.02	0	21.90	77.81	1.15	19.88	21.13
Spatial E	$\lambda = 0.4$	52.45	0	25.65	1.44	0	14.41	82.13	0.86	22.48	22.16
Classical D		85.30	0.29	39.77	1.15	0.29	17.58	49.86	0.29	16.14	23.41
Spatial D	$\lambda = 0.4$	88.47	1.15	34.58	0.28	0.28	10.66	67.72	5.18	28.24	26.29
Spatial D+E	$\lambda = 0.4$	89.05	0	53.31	0.86	0	25.65	72.91	0	13.54	28.37

REFERENCES

Barrow, H., tenenbaum, J., Bolles, R., and Wolf, H. (1977). Parametric correspondence and chamfer matching: two new techniques for image matching. In *Joint conference on artificial intelligence*, pages 659–663.

Duchenne, O., Bach, F. R., Kweon, I.-S., and Ponce, J. (2009). A tensor-based algorithm for high-order graph matching. In *CVPR*, pages 1980–1987.

Farabet, C., Couprie, C., Najman, L., and LeCun, Y. (2012). Scene parsing with multiscale feature learning, purity trees, and optimal covers. In *ICML*.

Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part based models. *IEEE Transactions on PAMI*, 32(9):1627–1645.

Felzenszwalb, P. F. and Huttenlocher, D. P. (2005). Pictorial Structures for Object Recognition. *IJCV*, 61(1):55–79.

Ferrari, V., Marin-Jimenez, M., and Zisserman, A. (2008). Progressive search space reduction for human pose estimation. In *CVPR*, pages 1–8.

Fischler, M. and Bolles, R. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communcations of the ACM*, 24:381395.

Fischler, M. and Elschlager, R. (1973). The representation and matching of pictorial structures. *IEEE Transactions on Computer*, 22(1):6792.

Geman, S. and Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on PAMI*, 6(6):721–741.

Holt, B., Ong, E.-J., Cooper, H., and Bowden, R. (2011). Putting the pieces together: Connected poselets for human pose estimation. In *ICCV Workshop on Consumer Depth Cameras for Computer Vision*.

Kolmogorov, V. and Zabih, R. (2004). What energy functions can be minimized via graph cuts? *IEEE Transactions on PAMI*, 26(2):147–159.

Lepetit, V., Pilet, J., and Fua, P. (2004). Point matching as a classification problem for fast and robust object pose estimation. In *CVPR*, pages 244–250.

Shotton, J., Blake, A., and Cipolla, R. (2008). Multiscale categorical object recognition using contour fragments. *IEEE transactions on PAMI*, 30(7):1270–81.

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., and Blake, A. (2011). Real-time human pose recognition in parts from single depth images. In *CVPR*.

Sivic, J. and Zisserman, A. (2003). Video google: A text retrieval approach to object matching in videos. In *ICCV*, volume 2, pages 1470–1477.

Torresani, L., Kolmogorov, V., and Rother, C. (2008). Feature correspondence via graph matching: Models and global optimization. In *ECCV*, volume 2, pages 596–609.

Winn, J. and Shotton, J. (2006). The layout consistent random field for recognizing and segmenting partially occluded objects. In *CVPR*, volume 1, pages 37–44.