

Relevant Feature Selection for Human Pose Estimation and Localization in Cluttered Images

Ryuzo Okada* and Stefano Soatto

Computer Science Department, University of California, Los Angeles
{okada, soatto}@cs.ucla.edu

Abstract. We address the problem of estimating human body pose from a single image with cluttered background. We train multiple local linear regressors for estimating the 3D pose from a feature vector of gradient orientation histograms. Each linear regressor is capable of selecting relevant components of the feature vector depending on pose by training it on a pose cluster which is a subset of the training samples with similar pose. For discriminating the pose clusters, we use kernel Support Vector Machines (SVM) with pose-dependent feature selection. We achieve feature selection for kernel SVMs by estimating scale parameters of RBF kernel through minimization of the radius/margin bound, which is an upper bound of the expected generalization error, with efficient gradient descent. Human detection is also possible with these SVMs. Quantitative experiments show the effectiveness of pose-dependent feature selection to both human detection and pose estimation.

1 Introduction

Human detection and pose estimation have numerous applications such as automated surveillance, driver assistance for automobiles, human-computer interfaces, and are an active area of research. These tasks are challenging because clothing, lighting conditions, and pose change the appearance of humans significantly. To cope with this problem, an image descriptor based on histograms of oriented gradients (HOG), which are computed on a uniform grid of overlapping local patches, has been used as a feature vector successfully for both human detection [1,2,3] and human pose estimation [4,5,6]. The descriptor also encodes unwanted background clutter into the feature vector, some of the feature vector components are irrelevant to the task. It is well known that such irrelevant components increase the complexity of classifiers and regressors and decrease generalization capacity, and that feature selection techniques [7] can be used for solving this problem.

One important aspect of relevant feature selection in detecting humans and estimating their pose is the fact that what features are relevant depends on pose (Fig. 1). Feature components that are relevant for some poses, are irrelevant to others, and this aspect has not received much attention in the literature. To address this issue, we propose a piecewise linear regression method where multiple local linear regressors approximate the nonlinear mapping function from HOG-based feature vectors to 3D poses. We train the

* Current affiliation: Corporate R&D Center, Toshiba Corporation.



Fig. 1. Relevant feature depending on pose. The diagonal gradient orientation in the left figure (shown by the diagonal white line segment in the white rectangle) is a relevant component of the feature vector for describing the pose of the right arm while the same component is irrelevant to the pose shown in the right figure.

local linear regressor for each pose cluster which contains a subset of training samples with similar poses. Since the linear regressor implicitly performs feature selection [4], relevant features are selected automatically for each pose cluster. Note that piecewise regression approaches have been used for pose estimation in a different context [8,9], where multiple regressors are employed for describing one-to-many mapping from an image feature to 3D poses, e.g. multiple different poses have similar silhouettes, and such ambiguity is solved by taking temporal consistency into account. In our method, we discriminate the pose cluster from the others to select the linear regressor to be used for 3D pose estimation. For this purpose, we train a Support Vector Machine (SVM) with feature selection for each cluster, which enables pose-dependent feature selection.

This cluster discrimination process is applicable to human detection as well by adding non-human training samples and training SVMs to discriminate each pose cluster from the non-human ones.

We achieve feature selection for each SVM using a RBF kernel by estimating scale parameter of the RBF kernel for each component of the feature vector separately. This is a problem of hyperparameter estimation (or model selection) for SVMs, and is solved by minimizing the radius/margin (R/M) bound [10] based on a gradient descent method [11]. Although this feature selection method for SVMs has not been used for vision problems, it reduces generalization error effectively because the R/M bound is an approximation to an upper bound of the expected generalization error. Furthermore, we point to a new way to efficiently compute the gradient of the R/M bound.

Related works: Agarwal et al. [4] recover frontal poses of the upper human body by regression from the feature vector based on HOG encoded by non-negative matrix factorization (NMF) to suppress unwanted background. In their experiments, the error of pose prediction using NMF encoding is similar to the one obtained by linear regression without NMF encoding. This is because the linear regressor implicitly performs feature selection. Bissacco et al. [12] estimate human full body pose based on multi-dimensional boosting regression which enables relevant feature selection from a preselected set of Haar-like features. Shakhnarovich et al. [6] find k-NN samples by a fast parameter (pose) sensitive hashing algorithm and estimate pose by locally weighted regression using the k-NN samples. They assume that the background is simple and stationary and that the human body is segmented from the background. Many approaches have been proposed to deal with background clutter, from body parts detection [13,14,15,16],

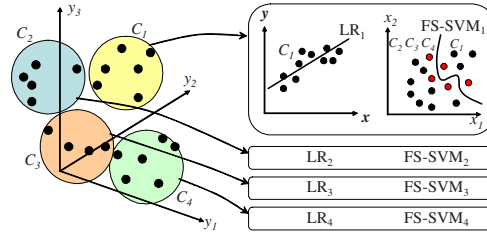


Fig. 2. Training piecewise linear regressors. The pose space is divided into several clusters by k-means. For each cluster, a linear regressor and an SVM are trained with feature selection.

which is particularly difficult in a single view because of self-occlusion, to silhouette extraction [8,17,18,19], which limits the applicability to fixed background, and edge-based template matching [9], which may be unstable under significant background clutter.

For Human detection, AdaBoost is used as a classifier to select relevant features from a set of, e.g., Haar-like wavelet features [20], gradient response in several directions [21], and SIFT-like gradient orientation features [2,3]. Support Vector Machines (SVMs) are also used for human detection with implicit feature selection (linear SVM) [1], and “filter-type” feature selection methods which preselect relevant features independently of the SVMs based on another machine learning method capable of feature selection, such as AdaBoost [22], or based on a heuristic that relevant features move decision boundaries significantly when they are removed [23]. Although the feature selection method is different and pose estimation is very coarse (front/left/right), [22] is similar to our approach in that it discriminates the pose clusters based on pose-dependent feature selection.

2 Piecewise Linear Regression

For recovering 3D human pose from a static image, a regression approach has been proposed in [8,4,17], where a mapping function from a feature vector extracted from the static image to a pose vector is learned using a set of labeled training samples. We take this type of regression based approach using the feature vector based on the histograms of oriented gradients (HOG) [4,5].

The feature vector consists of the histograms computed on a uniform grid of overlapping local patches to describe the contents of an image window. The histogram for each image patch encodes local shape and position information, while the coarse grid and orientation histogram is insensitive to variation in appearance and small misalignments. This representation also encodes unwanted background clutter into the feature vector and some of the components are irrelevant to human pose. More importantly, the relevant component is dependent on pose as shown in Fig. 1 because the relative position of the arms, legs, and torso in the image window vary depending on pose.

Considering this fact, we propose a piecewise linear regression method for recovering 3D pose from a feature vector $\mathbf{x} \in R^L$ (see Fig. 2). In this paper, we represent the

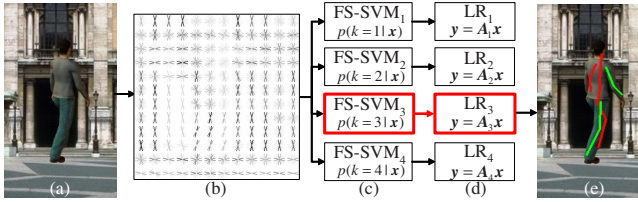


Fig. 3. Overview of our method for predicting 3D pose. (a) Given a test image, (b) the feature vector is computed in the window. (c) The pose cluster which the current pose belongs to is determined by SVM classifiers. (d) The 3D pose is recovered using the linear regressor of the selected cluster ($k=3$). (e) The recovered pose is shown by red and green line segments, where green represents the right leg and arm.

3D pose by a pose vector $\mathbf{y} \in R^P$ which is a collection of 3D locations of major joints. Given a set of training samples $\{(\mathbf{y}_i, \mathbf{x}_i) | i = 1 \cdots N\}$, we divide them into several clusters in the pose space by k-means so that each cluster contains similar poses, and train a linear regressor for each cluster C_k , which maps a feature vector \mathbf{x} to a pose vector \mathbf{y} :

$$\mathbf{y} = \mathbf{A}_k \mathbf{x} + \epsilon_k, \quad (1)$$

where \mathbf{A}_k and ϵ_k are a weight matrix and a residual error vector for a cluster C_k , respectively. It is important for our approach of pose-dependent feature selection that each pose cluster contains similar poses. We verified by preliminary experiments that each pose cluster generated by k-means (e.g. 6 clusters for walking sequences) was already local enough for approximating the nonlinear mapping function by multiple linear regressors, and that Expectation-Maximization type algorithms [9], which simultaneously optimize regressors and partition of clusters, did not improve generalization performance significantly. Although a kernel-based nonlinear regressor can be used with feature selection, we choose to use the linear regressor because of the following reasons: (1) The linear regressor is capable of implicit feature selection in the sense that it automatically reduces the weights with respect to irrelevant features through the estimation of the weight matrix \mathbf{A}_k . (2) The relation between the pose vectors and the feature vectors within a cluster is simple enough to be described by a linear function. (3) Training is much simpler than the kernel regressors with feature selection. Since similar poses in each cluster share the relevant features, each linear regressor achieves pose-dependent feature selection. Next, for each cluster, we train a Support Vector Machine (SVM) [10] with capabilities of feature selection [11] and probability output [24] in order to discriminate the cluster from the others. Feature selection is useful for discriminating the clusters as well because the feature vector contains cluttered background and the relevant features vary depending on the cluster to be discriminated.

For predicting a pose given a test image with a window circumscribing a subject, we first extract a HOG-based feature vector of the window (see Fig. 3). Secondly, we determine the cluster that the current pose of the subject belongs to by selecting the cluster with the highest probability output $p(k|\mathbf{x})$ of the SVM classifier. Thirdly, we predict the 3D pose using the linear regressor of the selected cluster.

2.1 Linear Regression Method

To train a linear regressor which recovers a pose vector \mathbf{y} from a feature vector \mathbf{x} , we estimate the weight matrix \mathbf{A}_k for each cluster C_k using training samples in C_k by minimizing prediction error with a regularization term $R(\cdot)$ to control overfitting:

$$\mathbf{A}_k = \arg \min_{\mathbf{A}_k} \left\{ \sum_{(\mathbf{y}_i, \mathbf{x}_i) \in C_k} \|\mathbf{A}_k \mathbf{x}_i - \mathbf{y}_i\|^2 + R(\mathbf{A}_k) \right\}. \quad (2)$$

This is equivalent to the MAP estimation in the probabilistic regression framework assuming a Gaussian distribution on the residual error vector. The first term, the data fidelity term, in the objective function corresponds to the likelihood, and the regularization term corresponds to the prior $p(\mathbf{A}_k)$.

For a Gaussian prior $p(\mathbf{A}_k) \sim \prod_l \exp(-\nu \|\mathbf{a}_l\|^2)$, where \mathbf{a}_l denotes the l -th column vector of \mathbf{A}_k , the regularizer takes the form $R(\mathbf{A}_k) \equiv \lambda \|\mathbf{A}_k\|_F^2$ and the solution gives a linear ridge regressor. The ridge regressor is capable of performing feature selection in the sense that the columns $\|\mathbf{a}_l\|$ weigh irrelevant components l , ideally belonging only to the background, automatically reducing their effect through the optimization (2).

Relevance Vector Machine (RVM) regression [25] results in taking a prior of the form $p(\mathbf{A}_k) \sim \prod_l \|\mathbf{a}_l\|^{-\nu}$, which gives a sparse solution because the prior is sharply peaked at $\|\mathbf{a}_l\| = 0$ and pushes the weights of the irrelevant components to zero. For achieving sparsity, a Laplace prior of the form $p(\mathbf{A}_k) \sim \prod_l \exp(-\nu \|\mathbf{a}_l\|)$ is often used and an ϵ -insensitive loss function is used as the data fidelity term in SVM regression [26]. In this paper, we use the RVM regressor because it gives a more sparse solution while maintaining good generalization performance. Note that the SVM regressor can produce sparse solutions comparable to the RVM regressor by employing ‘‘reduced-set’’ post-processing [27].

Since similar poses contained in each cluster share relevant features, the linear regressor is capable of selecting such relevant features depending on the poses in the cluster.

2.2 Cluster Discrimination Using SVM

In this section, we describe the SVM classifiers for discriminating a cluster C_k from the other clusters. For simplicity of description, we drop the subscript k referring to the cluster C_k in the rest of this section.

Feature selection is useful for discriminating the clusters for the same reason as in the regression case. We introduce feature selection into the kernel SVM based on automatic relevance determination (ARD), which is achieved by tuning the scale parameters $\gamma \in R^L \geq \mathbf{0}$ of the ARD Gaussian kernel

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp \left(-\frac{1}{2} \sum_l \gamma_l |x_{il} - x_{jl}|^2 \right). \quad (3)$$

Although a linear SVM is capable of implicit feature selection, the kernel SVM performs better than the linear SVM in our experiments. Tuning the kernel scale parameters

γ , which are hyperparameters, is achieved by minimizing an estimate of the generalization error such as the leave-one-out (LOO) error. The LOO error has an upper bound referred to as the radius/margin (R/M) bound [10].

$$\gamma = \arg \min_{\gamma} \frac{1}{N} R^2 \|\mathbf{w}\|^2, \tag{4}$$

where $2/\|\mathbf{w}\|$ is a margin between the classes to be discriminated and R is the radius of the smallest sphere that contains all the vectors $\mathbf{z}_i = \phi(\mathbf{x}_i)$ in a projected (high dimensional) feature space. Here the inner product in \mathbf{z} -space is computed by the kernel function: $(\mathbf{z}_i, \mathbf{z}_j) = K(\mathbf{x}_i, \mathbf{x}_j)$. This means tuning the hyperparameters that maximize the margin while making the sphere as small as possible.

Let $t_i \in \{-1, +1\}$ be a target value. $t_i = +1$ denotes a training sample $(\mathbf{y}_i, \mathbf{x}_i)$ is in the cluster C_k and otherwise $t_i = -1$. \mathbf{w} is the solution of the following SVM problem with hard margin¹:

$$\min \frac{1}{2} \|\mathbf{w}\|^2 \quad \text{s.t.} \quad t_i(\mathbf{w}^T \mathbf{z}_i + b) \leq 1 \quad \forall i. \tag{5}$$

R is the solution of the following problem:

$$\min R^2 \quad \text{s.t.} \quad \|\mathbf{z}_i - \mathbf{c}\| < R^2 \quad \forall i. \tag{6}$$

To find the optimum value of γ by minimizing the R/M bound, gradient descent is shown to be an efficient method [11]. We eliminate irrelevant features that have a small value of γ_l during the optimization process of (4) every 5 iterations of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) quasi-Newton algorithm [28].

To compute the derivative of the R/M bound, we need the derivatives of $\|\mathbf{w}\|^2$ and R^2 with respect to every scale parameter γ_l . This requires computing the entire kernel matrix for each γ_l (or storing it in memory to avoid re-computation), as done in [11,29].

Remark 1. The derivative of the margin $\|\mathbf{w}\|^2$ with respect to the scale parameter γ_l is computed using the support vectors only because the Lagrange multipliers α_i of the solution (5) are non-zero only for the support vectors:

$$\frac{\partial \|\mathbf{w}\|^2}{\partial \gamma_l} = \frac{1}{2} \sum_{\{i|\alpha_i \neq 0\}} \sum_{\{j|\alpha_j \neq 0\}} \alpha_i \alpha_j t_i t_j K(\mathbf{x}_i, \mathbf{x}_j) |x_{il} - x_{jl}|^2. \tag{7}$$

The derivatives of the radius R^2 can be computed similarly using the feature vectors corresponding to the non-zero Lagrange multipliers β_i of the solution (6).

Although this remark is not mentioned in the original papers [11,29], it reduces computational cost significantly, i.e., the computational complexity of (7) is $O(LN_{sv}^2)$ while that of the original formulation is $O(LN^2)$, where N , N_{sv} , and L are the numbers of training samples, support vectors, and feature dimensions, respectively. Hence the

¹ A non-separable case can be treated as an SVM problem with hard margins by converting the SVM problem using L_2 soft margin [11].



Fig. 4. Training images. The training images are synchronized with pose data. The synthesized dataset consists of cluttered background images (left) and clean background images (middle), where the white dots shows 13 joint locations used for encoding 3D pose. The real dataset (right) includes three subjects.

computational cost of (7) is smaller than that of the original formulation by a factor of $(N_{sv}/N)^2$.

To obtain the posterior probability $p(k|\mathbf{x})$ that the pose represented by a feature \mathbf{x} belongs to the cluster with index k , the mapping from the SVM output to the posterior probability is approximated by a sigmoid function [24]. The parameters of the sigmoid function are tuned by cross validation. Given a feature vector \mathbf{x} extracted from a test image, we compute the probability $p(k|\mathbf{x})$ for each cluster and determine that the pose represented by a feature \mathbf{x} belongs to the cluster with the highest probability.

2.3 Human Detection

The framework presented thus far for pose discrimination can be applied to human detection as well, by adding the non-human training samples. They are negative samples for training each of the SVMs: The target value is $t_i = +1$ for the training sample $(\mathbf{y}_i, \mathbf{x}_i)$ in the cluster C_k , and $t_i = -1$ for the non-human samples.

3 Experiments

Pose prediction: We have conducted experiments on a synthetic database and a real database (see Fig. 4). In the synthetic dataset, human poses are randomly generated in a subspace constructed by PCA using the walking sequences extracted from the CMU Motion Capture Database². Human images corresponding to each pose are rendered by a human model rendering software, Poser, with cluttered background of natural images and with uniform background. For real dataset, we used HumanEva-I dataset [30].

The human pose is represented by a 39-dimensional pose vector \mathbf{y} which is a collection of 13 major joint locations as shown in Fig. 4 (middle). The feature vector is histograms of oriented gradients. We compute the orientation of gradients in $[0, \pi]$ and construct the histograms using 8 orientation bins in 3×3 spatial cells which comprise a spatial block. We used uniformly spaced 4×4 blocks overlapping with neighbor blocks by the length of a cell and obtain a 1152 dimensional feature vector for a image window.

Fig. 5 shows pose prediction errors with respect to different regression methods on the synthetic dataset with clean and cluttered background. The prediction errors

² <http://mocap.cs.cmu.edu>

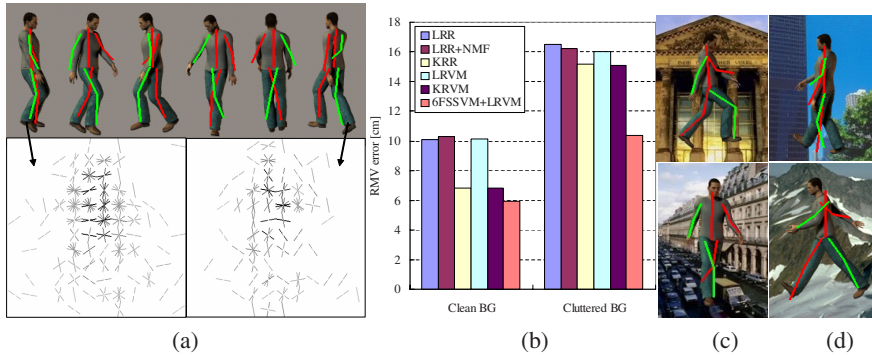


Fig. 5. Results of pose estimation on the synthetic dataset. (a) Mean poses of 6 pose clusters drawn by red and green (right arm and leg) line segments on the closest training image, and relevant components of the feature vector for the left and right pose clusters (components with dark color have high weights $\|a_l\|$). (b) Pose prediction errors with respect to different regression methods on clean and cluttered background. (c) Example of successful pose estimates. (d) Typical example of misestimation.

are measured in terms of the RMS deviation from the ground truth for the 3D locations of the 13 major joints. The number of training and test samples are 4000 and 1000, respectively. In the case of cluttered background, the prediction errors with respect to any single regressor (linear ridge regressor (LRR), LRR with NMF encoding (NMF+LRR) [4], RBF-kernel ridge regressor (KRR), linear RVM regressor (LRVM), and RBF-kernel RVM regressor (KRVM)) are equally large. Since our method is capable of reducing the disturbance of background clutter (see Fig. 5(a) bottom) by pose-dependent feature selection using 6 linear RVM regressors and 6 ARD kernel SVM classifiers³ (6FSSVM+LRVM), the prediction error of our method is almost as good as that for the clean background with a linear regressor (LRR or LRVM). The problem of multiple posterior modes in mapping from the HOG-feature to the 3-D pose is not severe on clean backgrounds. However, such multi-modality is a typical failure mode in the presence of cluttered background as shown in Fig. 5 (bottom right), where the left arm and leg are misestimated as the right ones. Note that the prediction error of 6FSSVM+LRVM for clean background is smaller than those of the nonlinear regression methods. This shows that the number of pose clusters, which we have determined to be 6 experimentally, are sufficient for approximating the nonlinear mapping function from HOG feature to 3D pose.

Fig. 6 shows pose prediction errors and sample pose estimates for the real image dataset of HumanEva-I. In this dataset, image sequences captured by calibrated cameras are synchronized with motion capture data. We used walking sequences of three subjects, S1, S2 and S3, captured by a camera C1 only. The training subset of the dataset contains 590, 438, and 435 frames for S1, S2, and S3, respectively. We add feature vectors slightly shifted in position and scale to increase tolerance to misalignment of the test window, and the total number of training samples is 4389 to train a

³ LIBSVM [31] is used with modification for solving the problems (5) and (6).

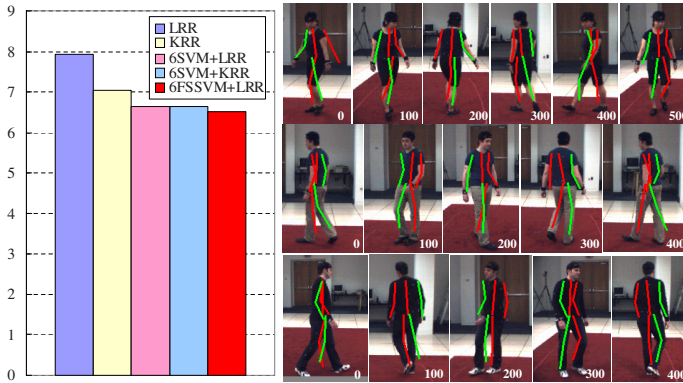


Fig. 6. Results of pose estimation on HumanEva-I dataset. The left bar graph shows pose prediction errors with respect to different regression methods. The right images show samples of predicted poses for 3 subjects drawn in the red and green (right arm and leg) line segments. (The head top joint (headDistal) is not used because of error in the dataset.)

Table 1. Mean errors of relative joint positions on the test subset of HumanEva-I dataset. The mean errors of our method without using background subtraction are smaller than those of Poppe’s method [5] based on background subtraction for the subject S1, S2 and total average.

Method	S1	S2	S3	Average [mm]
Poppe [5]	41.24	39.56	55.27	42.85
Ours	41.19	35.03	37.69	37.98

linear ridge regressor (LRR), an RBF-kernel ridge regressor (KRR), 6 SVMs with LRRs (6SVM+LRR) or KRRs (6SVM+KRR), and 6 feature-selection SVMs with LRRs (6FSSVM+LRR). For testing, we used the validation subset (ground truth is known) and the test window circumscribing the subject is given based on the known 3D position of the subject. Pose-dependent feature selection by 6SVM+LRR decreases the prediction error. It is competitive with 6SVM+KRR where feature selection is not performed. The prediction error further decreases by introducing pose-dependent feature selection for both SVMs and regressors (6FSSVM+LRR). Although the background in the test window varies according to the 3D location of the subject, improvement by 6FSSVM+LRR is limited because the background clutter in HumanEva-I dataset is much simpler than that of the synthetic dataset. Training the kernel SVMs with feature selection requires solving the problems (5) and (6) many times, and it took about 3 hours for each SVM in this experiments using HumanEva-I dataset on a standard PC with 2.13 GHz Intel CPU and 2 GB memory. Training time for each linear RVM regressor was about 5 seconds. For estimating the pose given a feature vector, it takes 25 ms on average.

We compare our method with the recent method proposed by Poppe [5] using HOG-based feature vector on the test subset of HumanEva-I dataset. Training data used in this experiment are the same as [5]. His method uses background subtraction to locate

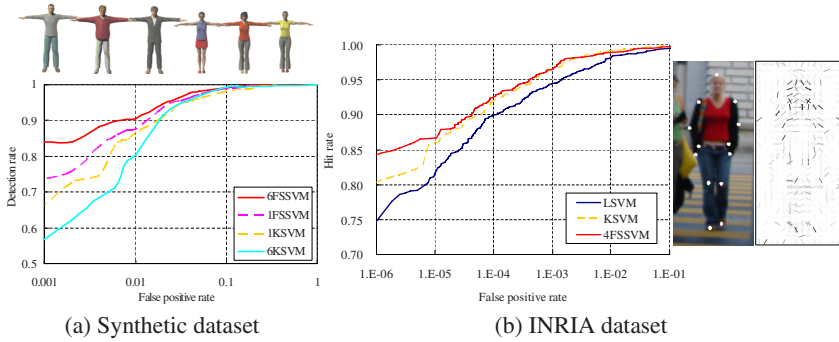


Fig. 7. Results of human detection. (a) Human models used for generating training and test images (top), and ROC curves of different classifiers (bottom). (b) ROC curves of our detector and the linear SVM [1] (left). The mean pose of a pose cluster is drawn on the closest training image (middle). Pose is represented by 13 points which are shown by white dots. Estimated kernel scale parameters γ_l for each feature component are shown by short line segments (right). Black color means a high value of γ_l .

the subject accurately in each image while our method uses the SVM-based human detector described in section 2.3, whose negative training samples are randomly sampled image patches without overlapping with human region more than 50%. Although we can search the entire image for obtaining global 3D position of the subject, we apply our human detector locally on a $5 \times 5 \times 7$ grid covering $40 \times 40 \times 150$ cm (wide search range in the camera direction) around the previous global position to reduce the computational cost. We obtain the current 3D global position as the mean of the weighted positions where the human detector is applied. The weights are the probability outputs of the human detector. In the initial frame, we give a rough estimate of 3D global position by watching the image since we do not know the ground truth in the test subset. Table 1 shows the mean errors of joint positions relative to the pelvis (torsoDistal) joint. Although our method does not perform background subtraction, the mean errors of our method are smaller than those of Poppe’s method.

Human detection: Fig. 7(a) shows results of human detection using the synthetic dataset consisting of 6 subjects with different clothing and poses. The pose variation is the same as the case of pose estimation and each subject has 1000 samples. We use 5000 samples of the left 5 subjects in Fig. 7(a) and 1500 non-human samples for training. For testing, we use 1000 samples of the subject on the right and 1000 samples of non-human images which are not included in the training samples. Introducing the feature selection on a single SVM (1FSSVM) improves the performance and pose-dependent feature selection using 6 SVMs (6FSSVM) outperforms the other methods. The prediction error for the method using 6 kernel SVMs (6KSVM) is worse than 1KSVM due to the lack of samples for training each of the 6 kernel SVMs, which are much fewer than those of 1KSVM.

We test our human detector on the real images of the INRIA person dataset [1]⁴ to illustrate the flexibility of our approach that uses pose-dependent relevant features. Since

⁴ <http://pascal.inrialpes.fr/data/human/>

our method requires pose information to build pose clusters, we added pose information to the positive (human) images by locating the 13 positions, such as head, elbows, knees and so on, by hand (see Fig. 7(b) middle). We trained our detector using 4 pose clusters, a linear SVM and a kernel SVM with the default setting of [1] on 1,246 positive and randomly sampled 12,180 negative samples plus hard negative samples for retraining, which are false positives in the first training. Fig. 7(b) shows the ROC curves and an example of estimated kernel scale parameters for a pose cluster. Our detector improves performance by 12% and 8% at 10^{-6} false positive rate compared with linear SVM and kernel SVM, respectively.

4 Conclusions

We presented a method for human detection and pose estimation from a static image with background clutter. We estimated the 3D pose from a feature vector consisting of local gradient orientation histograms by first discriminating the pose clusters using the kernel SVMs with feature selection and then estimating the pose using linear regressors. We performed feature selection for kernel SVMs by estimating scale parameters of ARD RBF kernel through minimization of the R/M bound which is an upper bound of the expected generalization error. For minimizing the R/M bound with gradient descent, we computed its gradient efficiently by using support vectors only. The process of discriminating the pose clusters by using the SVMs is applicable for human detection by training the SVMs with non-human samples. Both SVMs and linear regressors are capable of pose-dependent feature selection, which was shown to be effective by the quantitative experiments where our method was compared with other recent methods and outperformed them.

References

1. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. of CVPR, vol. 2, pp. 886–893 (2006)
2. Mikolajczyk, K., Schmid, C., Zisserman, A.: Human detection based on a probabilistic assembly of robust part detectors. In: Pajdla, T., Matas, J.(G.) (eds.) ECCV 2004. LNCS, vol. 3021, pp. 69–81. Springer, Heidelberg (2004)
3. Zhu, Q., Avidan, S., Yeh, M.C., Cheng, K.T.: Fast human detection using a cascade of histograms of oriented gradients. In: Proc. of CVPR, vol. 2, pp. 1491–1498 (2006)
4. Agarwal, A., Triggs, B.: A local basis representation for estimating human pose from cluttered images. In: Proc. of ACCV, vol. 1, pp. 50–59 (2006)
5. Poppe, R.: Evaluating example-based pose estimation: experiments on the HumanEva sets. In: Computer Vision and Pattern Recognition (CVPR 2007) workshop on Evaluation of Articulated Human Motion and Pose Estimation (EHuM2) (2007)
6. Shakhnarovich, G., Viola, P., Darrell, T.: Fast pose estimation with parameter-sensitive hashing. In: Proc. of ICCV, vol. 2, pp. 750–757 (2007)
7. Blum, A., Langley, P.: Selection of relevant features and examples in machine learning. *Artificial Intelligence* 97(1-2), 245–271 (1997)
8. Agarwal, A., Triggs, B.: Monocular human motion capture with a mixture of regressors. In: IEEE Workshop on Vision for Human-Computer Interaction, pp. 1–8 (2005)

9. Thayananthan, A., Navaratnam, R., Stenger, B., Torr, P.H.S., Cipolla, R.: Multivariate relevance vector machines for tracking. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3953, pp. 124–138. Springer, Heidelberg (2006)
10. Vapnik, V.N.: *Statistical Learning Theory*. Wiley-Interscience, Chichester (1998)
11. Keerthi, S.S.: Efficient tuning of SVM hyperparameters using radius/margin bound and iterative algorithms. *IEEE Trans. on Neural Networks* 13(5), 1225–1229 (2002)
12. Bissacco, A., Yang, M.H., Soatto, S.: Fast human pose estimation using appearance and motion via multi-dimensional boosting regression. In: *Proc. of CVPR*, pp. 1–8 (2007)
13. Date, N., et al.: Real-time human motion sensing based on vision-based inverse kinematics for interactive applications. In: *Proc. of ICPR*, vol. 3, pp. 318–321 (2004)
14. Sigal, L., Bhatia, S., Roth, S., Black, M.J., Isard, M.: Tracking loose-limbed people. In: *Proc. of CVPR*, vol. I, pp. 421–428 (2004)
15. Sigal, L., Black, M.J.: Predicting 3D people from 2D pictures. In: *Proc. of Conf. Articulated Motion and Deformable Objects*, pp. 185–195 (2006)
16. Wren, C.R., Azarbayejani, A., Darrell, T., Pentland, A.P.: Pfunder: Real-time tracking of the human body. *IEEE Trans. on PAMI* 19(7), 780–785 (1997)
17. Agarwal, A., Triggs, B.: Recovering 3D human pose from monocular images. *IEEE Trans. on PAMI* 28(1), 44–58 (2006)
18. Okada, R., Stenger, B., Kondoh, N.: A video motion capture system for interactive games. In: *Proc. of MVA*, pp. 186–189 (2007)
19. Sminchisescu, C., Kanaujia, A., Li, Z., Metaxas, D.: Learning to reconstruct 3D human motion from bayesian mixtures of experts. a probabilistic discriminative approach. Technical Report CSRG-502, University of Toronto (2004)
20. Viola, P., Jones, M., Snow, D.: Detecting pedestrians using patterns of motion and appearance. In: *Proc. of ICCV*, pp. 734–741 (2003)
21. Sabzmejdani, P., Mori, G.: Detecting pedestrians by learning shapelet features. In: *Proc. of CVPR*, pp. 1–8 (2007)
22. Zehnder, P., Koller-Meier, E., Gool, L.V.: A hierarchical system for recognition, tracking and pose estimation. In: Bengio, S., Bourlard, H. (eds.) *MLMI 2004*. LNCS, vol. 3361, pp. 329–340. Springer, Heidelberg (2005)
23. Evgeniou, T., Pontil, M., Papageorgiou, C., Poggio, T.: Image representations for object detection using kernel classifiers. In: *Proc. of ACCV*, pp. 687–692 (2000)
24. Platt, J.: Probabilistic outputs for support vector machines and comparison to regularize likelihood methods. In: Smola, A., Bartlett, P., Schoelkopf, B., Schuurmans, D. (eds.) *Advances in Large Margin Classifiers*, pp. 61–74 (2000)
25. Tipping, M.E.: Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research* 1, 211–244 (2001)
26. Micchelli, C.A., Pontil, M.A.: On learning vector-valued functions. *Neural Computation* 17(1), 177–204 (2005)
27. Scholkopf, B., Smola, A.J.: *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge (2001)
28. Shanno, D.F., Phua, K.H.: Minimization of unconstrained multivariate functions. *ACM Transactions on Mathematical Software* 6, 618–622 (1980)
29. Chapelle, O., Vapnik, V., Bousquet, O., Mukherjee, S.: Choosing multiple parameters for support vector machines. *Machine Learning* 46, 131–159 (2002)
30. Sigal, L., Black, M.J.: Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical Report CS-06-08, Brown Univ. (2006), <http://vision.cs.brown.edu/humaneva/>
31. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001), <http://www.csie.ntu.edu.tw/~cjlin/libsvm>