# Learning Generic Human Body Models [⋆]

Thomas Walther and Rolf P. Würtz

Institut für Neuroinformatik, Ruhr-Universität, 44780 Bochum, Germany
thomas.walther@ini.rub.de, rolf.wuertz@ini.rub.de

**Abstract.** We describe a posture estimation system based on Organic Computing concepts, which learns a generic body model from video input in a self-governed manner. We show experimentally that the constructed model generalizes well to different attire and persons.

## 1 Introduction

Analyzing human body poses by mere observation is a topic of growing interest in computer vision — with application potential ranging from surveillance over man-machine communication to motion picture animation. Yet, the artificial pose estimation (PE) approaches developed over the last two decades are nowhere close to matching human visual skills. This may be due to different working principles of artificial and biological vision systems. In the following, we aim at levelling these differences by Organic Computing (OC) concepts, in short, the attempt to make artificial systems more self-organized in their behavior [1]. In particular, we propose a PE system that acquires knowledge in a completely unsupervised manner directly from video input; this knowledge is then generalized to novel situations, mimicking human skills in 'non-trivial' and continuous learning [2]. We build on work done by [3,4] to assemble autonomously acquired visual data into a higher-level *meta model* for the acquired knowledge. After training on videos of a moving human's upper body the resulting model is shown to generalize well to different movements, attire, and individuals.

## 2 Method

In the following, we assume a *segmentation method* that reliably extracts non-rigid upper human body parts in a completely autonomous manner from simple, fronto-parallel, monocular input streams. The method is further presumed to extract connections between the single limbs (the upper body skeleton) coevally and to learn the distribution of relative body joint angles. We ignore the neck joint here, as rotational motion orthogonal to the image plane is hard to capture in a monocular setting and significant in-plane motion of the head relative to the torso is rare. Such a system has been proposed by [3] and [4]; other approaches (e. g. [5]) could, with modifications, also be employed for non-rigid limb segmentation and skeleton construction.

[⋆] Funding by the DFG (WU 314/5-2, WU 314/5-3) is gratefully acknowledged.

Let $\mathcal{M}_q = \{\mathcal{L}_q, \mathcal{J}_q, \mathcal{D}_q\}$ describe an upper body model extracted from input sequence $q$, where $\mathcal{L}_q = \left\{\mathbf{L}_q^l\right\}$ with $l = 0 \ldots N_L - 1$ represents the $N_L$ body part appearance templates acquired from sequence $q$. Information concerning the kinematic skeleton structure of $\mathcal{M}_q$ (relative joint locations, connectivity) is stored in $\mathcal{J}_q$. Eventually, the distribution of relative joint angles for each skeleton joint (learned from all frames of sequence $q$) is stored in $\mathcal{D}_q$. Appearance models are retrieved from all input frames between an initial frame $f_B$ and a stop frame $f_E$. Then each $\mathbf{L}_q^l$ holds a separate appearance representation of limb $l$ for each valid frame $f$ of video stream $q$, such that $\mathbf{L}_q^l = \left\{\mathbf{l}_{q,f}^l\right\}, \ f \in [f_B, f_E]$. Moreover, $\mathbf{p}_{q,f}^l$ contains the pose (x, y, orientation, scale) of limb $l$ in sequence $q$ for frame $f$; stored in world coordinates. By letting $\mathbf{l}_{q,f}^l = \left\{\mathbf{s}_{q,f}^l, \mathbf{c}_{q,f}^l\right\}$, we point out separation of limb appearance templates into a shape map $\mathbf{s}_{q,f}^l$ and an RGB color map $\mathbf{c}_{q,f}^l$ defined for each valid input frame of sequence $q$. The shape map with values in [0,1] measures the relevance of each pixel to the limb's shape.

Assume that limb segmentation is applied to $N_Q$ input video sequences, resulting in a data set $\mathfrak{M} = \{\mathcal{M}_0, \ldots, \mathcal{M}_{N_M-1}\}$ of $N_M = N_Q$ separate upper body models, which differ significantly w.r.t. clothing, motion patterns and slightly w.r.t. illumination. Self-occlusion of the limbs and variation of the depicted subject are not allowed in this segmentation stage of the learning algorithm.

In the following, we consolidate the models in data set $\mathfrak{M}$ into a single *meta model*, that represents the upper human body on a more abstract level while preserving *pertinent features* that characterize human appearance. Such a meta model is predestined to show good generalization during matching: it focuses on salient features typical for human beings (mean limb outline, persistent color patches on head and hands), while generalizing well across meaningless details like cloth color and deformation, illumination, and motion patterns.

Meta model generation is based on two subprocesses: *intra-sequence limb prototype generation* and *inter-sequence limb prototype construction* ; borrowing from the biological paradigm [6], formulation of these prototypes is based on the evaluation of shape and color features in the input streams. Note that prototyping techniques are not unchallenged when it comes to body model construction and matching; [7] proposes, for instance, an interesting exemplar-based approach to detect animal or human body models in given image data. Furthermore, it is still discussed if human concept building capabilities foot on mental prototypes, exemplars or some different information management paradigm [8]. We decided in favor of the prototype approach here, as it principally allows to handle unlimited amounts of input data while keeping the memory footprint well-arranged and information retrieval times rather small.

## 2.1 Intra-sequence limb prototypes

Intra-sequence limb prototypes are rather straightforward to construct; for a dedicated limb $\hat{l}$ in input sequence $\hat{q}$, they unify the content of shape and color information memories $\mathbf{l}_{\hat{q},f}^{\hat{l}}$ from all valid frames $f = [f_B \ldots f_E]$.

**Shape prototypes** Formulating a shape prototype for a structure that deforms as vividly as a dressed limb is not trivial. Landmark-based methods, which are quite standard to derive mean shapes and deformation modes from deformable objects (e. g. *point distribution models* [9]) are not applicable in the current context, as landmark finding would have to rely on human intervention, thereby spoiling any previous attempts to maximize system autonomy. Further, automatic landmark finding procedures are, due to significant deformation of the body parts, not reliable enough to replace manual annotation. For these reasons, we choose a different approach to arrive at a fuzzy 'mean' shape of the observed limb templates; our method is based on *Gaussian voting* and remotely inspired by the approach presented in [10]; inherently capitalizing on knowledge of limb poses in each valid input frame.

For the following discussion, focus, without loss of generality, on a single limb $\hat{l}$ in a given sequence $\hat{q}$; it is quite natural to treat $\mathbf{p}^{\hat{l}}_{\hat{q},f_B}$ as the *reference pose* of the processed limb. With that, set up two different operators: first, let $G\left(\cdot\right)$ define a Gaussian blur operator with standard deviation $\sigma_B = 5.0$. Applying this operator to an arbitrary shape map $\mathbf{s}^{\hat{l}}_{\hat{q},f}$ dilutes the formerly crisp body part outline. Additionally, install a *registration operator* $R\left(\cdot\right)$ that projects limb shape map $\mathbf{s}^{\hat{l}}_{\hat{q},f}$ from any valid frame $f$ back to frame $f_B$.

Given this foundation, setting up the intra-sequence shape prototype ${\mathbf{s}^*}^{\hat{l}}_{\hat{q}}$ for limb $\hat{l}$ in sequence $\hat{q}$ can be formulated as a 'Gaussianized voting' procedure:

$$ {\mathbf{s}^*}^{\hat{l}}_{\hat{q}} = G\left(\mathbf{s}^{\hat{l}}_{\hat{q},f_B}\right) + \sum_{f=f_B+1}^{f_E} G\left(R\left(\mathbf{s}^{\hat{l}}_{\hat{q},f}\right)\right) \ . \tag{1} $$

While the 'voting' terminology had been lent from [10], the 'Gaussian' tag emphasizes our method of blurring the limb shapes prior to summation. This procedure to some degree compensates for the vivid cloth deformation behavior and results in smooth, naturally looking prototypical intra-sequence shapes. Eventually, the shape prototype is normalized (i.e., rescaled, such that the maximum summed voting value becomes 1.0), then the 25% weakest votes are removed to exclude spurious shape elements from further consideration. The resulting final intra-sequence shape prototype is re-normalized.

**Color prototypes** To arrive at the intra-sequence color prototypes, a different strategy is employed: first, given the reference pose $\mathbf{p}^{\hat{l}}_{\hat{q},f_B}$, reuse registration operator $R\left(\cdot\right)$ to project a limb color map $\mathbf{c}^{\hat{l}}_{\hat{q},f}$ from any valid frame $f$ back to frame $f_B$. Let the intra-sequence color prototype be

$$ {\mathbf{c}^*}^{\hat{l}}_{\hat{q}} = \frac{1}{(f_E - f_B + 1)} \left[ \mathbf{c}^{\hat{l}}_{\hat{q},f_B} + \sum_{f=f_B+1}^{f_E} R\left(\mathbf{c}^{\hat{l}}_{\hat{q},f}\right) \right] \ . \tag{2} $$

i.e., the intra-sequence color prototype for limb $\hat{l}$ is the mean of all sampled color observations for this body part. Note that this procedure necessarily blurs the prototype, due to slight tissue and more significant cloth deformation. Yet, this blur does not severely distort the fundamental color distribution of the prototypical limb and is tolerated henceforth.

Given the intra-sequence limb prototypes, we combine these relatively specialized descriptors into more abstract inter-sequence body part prototypes that show better generalization capabilities.

### 2.2 Inter-sequence limb prototypes

We now return to the meta model announced above: the limbs of this generic body description essentially represent the sought-after inter-sequence prototypes. To avoid notational confusion, let these limbs henceforth be termed *meta limbs*, whereas the joints of the meta model are termed *meta joints* from here on.

Initially, the meta limbs are instantiated with the shape/color prototypes of the *primary model* $\mathcal{M}_0$; also the meta joint structure (i.e., the skeleton of the meta model) is copied from the primary model. With that, define a procedure that aligns every *subsequent model* $\mathcal{M}_n,\ n = 1 \dots N_M - 1$ with the current meta model $\mathcal{M}_{\text{meta}}$. For simplicity, we focus on a single subsequent model $\mathcal{M}_{\hat{m}}$ in the following. The alignment procedure first performs simple model matching (based on routines described in [4] and section 2.4) to identify limb correspondences between the meta limbs and the body part prototypes in $\mathcal{M}_{\hat{m}}$. Using these results, the limbs of $\mathcal{M}_{\hat{m}}$ are eventually aligned with the meta limbs; further, the skeleton structure of the subsequent model is rearranged to coincide with the skeleton structure of the meta model. Note that during these processes, limb and joint characteristics (i.e., limb orientations, relative joint angles) of $\mathcal{M}_{\hat{m}}$ are appropriately adopted. With both models completely aligned, information from $\mathcal{M}_{\hat{m}}$ can be used to update the current meta limbs and the skeleton structure of $\mathcal{M}_{\text{meta}}$.

**Shape prototypes** To transfer limb appearance information from $\mathcal{M}_{\hat{m}}$ to the meta limbs, the approximate alignment established above is not sufficient; it, however, constitutes a good basis for further registration refinement: define an operator ICP $(\cdot)$ that applies the well-established 2D iterative closest point methods of [11] (accelerated according to [12]) to fine-register the limb shape prototypes of $\mathcal{M}_{\hat{m}}$ to their corresponding meta limbs. To keep computational effort at bay, we here perform shape registration on a thinned shape representation (thinning algorithm after [13]). With that, the inter-sequence shape prototype ${\mathbf{s}^*}^{\hat{l}}_{\text{meta}}$ for a certain meta limb $\hat{l}$ can be constructed from $N_Q$ input sequences as follows

$${\mathbf{s}^*}^{\hat{l}}_{\text{meta}} = {\mathbf{s}^*}^{\hat{l}}_{0} + \sum_{i=1}^{N_Q-1} \text{ICP}\left({\mathbf{s}^*}^{\hat{l}}_{i}\right), \tag{3}$$

i.e., the meta limbs fuse shape information from the single models by plain superposition of the previously learned, registered intra-sequence prototypes. Normalization (s. above) and removal of the 25% weakest votes yields the final

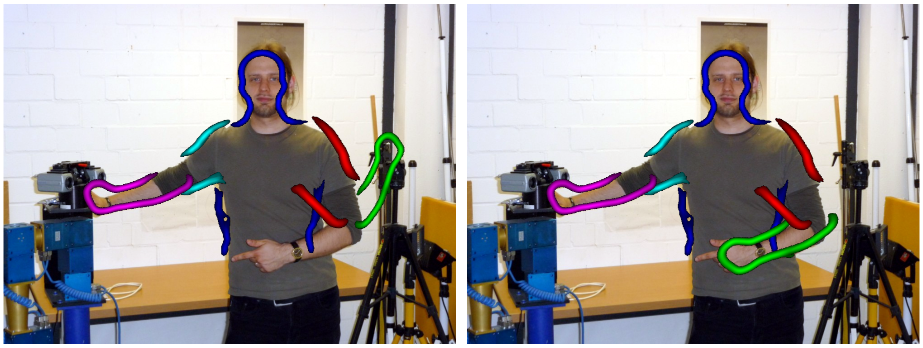

Fig. 1: Effect of limb flipping: the right image shows significantly better matching performance, as the left forearm is flipped orthogonally to the image plane.

meta limb shape. Obviously, this procedure favors stable shape parts (which persist throughout all input sequences), whereas cloth induced deformations are largely suppressed.

**Color prototypes** Deriving persistent color information from the captured models is somewhat more involved. First, assume that the above ICP operator results can be reused to register the color representation $\mathbf{c}^{*\hat{l}}_{\hat{q}}$ to its corresponding meta limb which had been learned from all sequences $0 \ldots \hat{q}-1$. Derive a binary *persistent color mask* $P^{\hat{l}}_{\hat{q}}(\mathbf{x})$ that takes on values of 1 where color features within the current meta limb and the registered intra-sequence prototype coincide. We construct this mask by performing a windowed (15×15 pixels window size), color histogram-based correlation, setting mask pixels $\mathbf{x}$ to zero whenever correlation scores drop below 0.25. The resulting mask is then slightly eroded to prevent learning from border sites. Note that the histogram correlation presumes the limb color maps to be given in HSV color space. Choice of this color space allows to exclude the value (V) component from further consideration, rendering histogram-based processing more robust w. r. t. illumination variations [14].

Using $P^{\hat{l}}_{\hat{q}}(\mathbf{x})$, a *color prototype accumulator* $\mathbf{c}^{\hat{l}}_{\text{acc}}$ is iteratively constructed from all models in $\mathfrak{M}$:

1. Primary model initialization:

$$\mathbf{c}^{\hat{l}}_{\text{acc}} \leftarrow \mathbf{c}^{*\hat{l}}_{0} \,.$$

2. For each subsequent model $\mathcal{M}_i \in \mathfrak{M} : (i = 1 \ldots N_M - 1)$

$$\mathbf{c}^{\hat{l}}_{\text{acc}}(\mathbf{x}) = \begin{cases} \mathbf{c}^{\hat{l}}_{\text{acc}}(\mathbf{x}) + \text{ICP}\left(\mathbf{c}^{*\hat{l}}_{i}\right)(\mathbf{x}) & \text{if } P^{\hat{l}}_{i}(\mathbf{x}) > 0 \\ 0 & \text{else} \end{cases} \,. \tag{4}$$

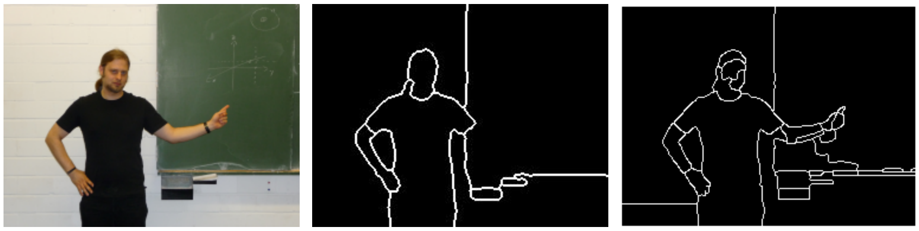

Fig. 2: Results of JSEG (center) and EDISON (right) edge segmenters on the image on the left; parameters are chosen according to [15] and [16], respectively.

At each iteration $i$, prototypical color information ${\mathbf{c}^{*}}^{\hat{l}}_{\text{meta}}$ for meta limb $\hat{l}$ can trivially be instantiated

$$ {\mathbf{c}^{*}}^{\hat{l}}_{\text{meta}} = \frac{\mathbf{c}^{\hat{l}}_{\text{acc}}}{i+1} \tag{5} $$

and used for determining $P^{\hat{l}}_{i+1}(\mathbf{x})$. Note that the above correlation threshold is quite generous, allowing for a significant number of 'false positive' persistent color regions to evolve during each model update cycle. However, by learning from multiple limb instances displaying vividly varying cloth colors, the true persistent color patches (e. g. hands and head) will eventually pop out.

### 2.3 Meta skeleton retrieval

Compared to the prototyping approaches used for shape and color features, skeleton prototyping is straightforward. Whereas the overall *meta skeleton* is necessarily identical to the primary model's skeleton w.r.t. connectivity, relative locations of the meta joints are found by averaging the relative joint locations from all $\mathcal{J}_i$, with $i = 0 \dots N_M - 1$. Similarly, the distribution of relative meta joint angles is learned by aggregating $\mathcal{D}_i$ for $i = 0 \dots N_M - 1$.

### 2.4 Meta model matching

To match the fully evolved meta model to novel input images, we employ a *pictorial structure* (PS) matching scheme similar to the one proposed by [17]. Due to the tree-like structure of the learned models, this dynamic programming approach allows to speed up model matching significantly while guaranteeing to yield globally optimal results. [4] gives an overview of the employed baseline scheme; here we enhance their approach in several aspects: first, we allow the matching algorithm to not only find the location (shift, rotation, scale) of each meta limb, but also to infer if a body part is flipped or not. This enables the system to cope with *kinematic flips* (terminology chosen in allusion to [18]). Such flips occur due to the 3D nature of the captured scenario and have to be taken into account to allow analysis of a broader range of body postures. It is assumed

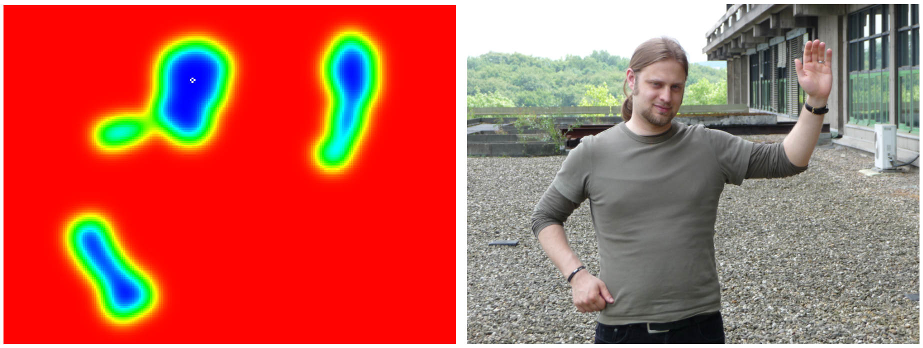

Fig. 3: Final color cue map produced by the left meta forearm for the input image on the right.

that each body part can only be flipped orthogonal to the image plane (around the limbs' major principal axis); the limits and angular statistics of each joint attached to the flipped body part are updated automatically. Fig. 1 clarifies the importance of flipping capabilities in our system.

Second, matching reliability is increased by refining the matching cost function constructed in [4]: shape matching cost is now computed using the *oriented Chamfer* distance (cf. [19]) between the meta limb shapes and a line segmentation of the given query image. The stand-alone JSEG [15] algorithm utilized in [4] to generate this line segmentation has been replaced by the EDISON [16] image segmentation scheme that is fully integrated into our system. Quality of line images generated by EDISON perceptually compares to or even outperforms the JSEG output (cf. fig. 2). Note that we outsource oriented Chamfer calculations to the GPU (using a CUDA-based implementation), to compensate for the increased computational effort inherent to this more powerful approach. To save computation time, the above fuzzy meta shapes are thinned (as above, thinning algorithm from [13]) prior to being used for oriented Chamfer matching.

Adding up to the above, it is straightforward to exploit the persistent color feature stored in each meta limb for derivation of a per-limb *color cue* map: for that, we first transform the RGB representation of the query image to HSV color space. Let then $\mathbf{W}(\mathbf{x})$ define a window (7×7 pixels) centered at position $\mathbf{x}$ in the HSV representation of the query image. Assume that an HS-histogram can be derived (during a batch-processing step not described here due to the page limit) from the meta limb's persistent color regions. A similar histogram is deemed available for the window patch. We again drop the V-component during histogram construction for better invariance to illumination variation. The map value at $\mathbf{x}$ is then calculated as the correlation of the two HS-histograms. To get rid of spurious elements, we apply a threshold of 0.1, and a Gaussian with $\sigma = 5.0$ is centered at each surviving map entry, to account for possible wrong negative color detections. Loosely following [20], the final color cue map is used (after inversion and re-scaling) to define an additional *color matching cost* that backs up the shape cue described above and renders overall matching behavior more robust. An exemplary color map is shown in fig. 3.

## 3 Experimental results and discussion

After learning from different sequences of one person, we matched the model into still images of different persons under different lighting conditions and with different backgrounds. The results in figure 4 demonstrate the generalization capabilities of the model. The system is able to produce good inference of body posture even in situations it had never been intended for and shows good generalization capabilities in the presence of significant background clutter, regardless of subject identity. So far, we have demonstrated the successful analysis of still images. A quantitative analysis on the basis of hand-annotated images is currently under way.

Several 2D approaches for human posture identification have been employed. In [21] a cue combination similar to ours is used to achieve robust limb matching from a manually trained model. [22] and also [23] present learning-based approaches for posture estimation based on pictorial structures with model initialization as well as body predetection based on human hand-crafting and domain knowledge. Further, spatio-temporal constraints are exploited to make posture recognition more reliable, which prevents their systems from analyzing still images. [24] strive to solve the pose estimation problem on single, 2D input images; their technique shows impressive capabilities, yet also relies on higher level domain knowledge provided by human supervisors. In contrast, our system autonomously achieves acquisition of similar knowledge (e.g., color cues or kinematic constraints). [5] learns body models (of humans and animals) with occlusions in a fully autonomous way from video input. Their approach could serve for limb segmentation in our framework, but does not extract an explicit skeleton, and the tuning of a significant number of parameters appears tedious. In [25], a pictorial structure model is learned from input data, while the input is already hand-labeled (contradicting OC ideas) and the learned PS model's rectangular shapes inevitably display less detail than our meta limbs.

The system proposed in this work complies with Organic Computing directives in that all required model information is generated autonomously; achieved generalization performance is good, as demonstrated experimentally. These encouraging results notwithstanding several improvements are required. Creation of the meta model depends on the order of video presentation, an effect that needs to be quantified and eliminated by appropriate modifications to the learning scheme. Blandly using the thinned meta shapes for oriented Chamfer matching may be problematic – at least a weighting scheme projecting circumjacent values from the fuzzy meta shape maps to the thinned limb boundary representation is required. We also plan to replace thinning by weighted spline techniques. Eventually, to veer away from pure theory, we will use our system to render a humanoid robotic device capable of understanding and mimicking human upper body motion.

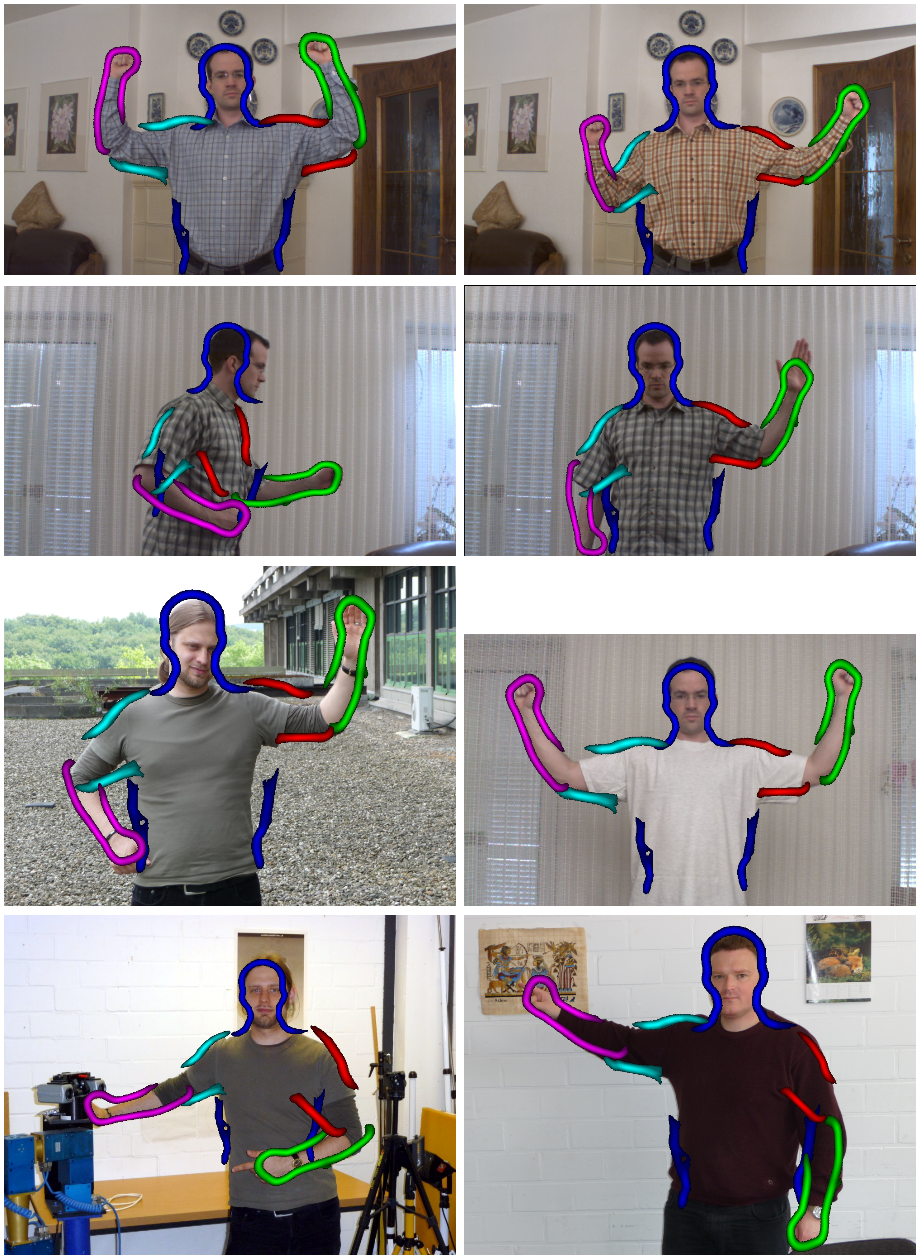

Fig. 4: Experimental results showing the range of applicability of the learned model (clockwise from upper left): Matching to the same person as in the model but wearing different shirts and with a variety of backgrounds and lighting conditions; different persons with different shirts and varying backgrounds; and finally, a side view, which was not seen at all during training.

## References

1. Würtz, R.P., ed.: Organic Computing. Springer (2008)
2. Poggio, T., Bizzi, E.: Generalization in vision and motor control. Nature **431** (2004) 768–774
3. Walther, T., Würtz, R.P.: Learning to look at humans - what are the parts of a moving body. In: Proc. AMDO, Springer (2008) 22–31
4. Walther, T., Würtz, R.P.: Unsupervised learning of human body parts from video footage. In: Proceedings of ICCV workshops, Kyoto, IEEE Computer Society, Los Alamitos, CA (2009) 336–343
5. Kumar, M.P., Torr, P.H.S., Zisserman, A.: Learning layered motion segmentations of video. International Journal of Computer Vision **76**(3) (2008) 301–319
6. Bear, M.F., Connors, B.W., Paradiso, M.A.: Neuroscience – Exploring the Brain 3rd Edition. Lippinscott Williams & Wilkins (2006)
7. Kumar, M.P., Torr, P.H.S., Zisserman, A.: Objcut: Efficient segmentation using top-down and bottom-up cues. IEEE Trans. PAMI **32**(3) (January 2009) 530–545
8. Murphy, G.L.: The Big Book of Concepts. The MIT Press (2004)
9. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models—their training and application. Comput. Vis. Image Underst. **61**(1) (1995) 38–59
10. Lee, Y.J., Grauman, K.: Shape discovery from unlabeled image collections. In: Proc. CVPR, IEEE (2009) 2254–2261
11. Besl, P.J., McKay, N.D.: A method for registration of 3-D shapes. IEEE Trans. PAMI **14**(2) (1992) 239–256
12. Rusinkiewicz, S., Levoy, M.: Efficient variants of the ICP algorithm. In: Proc. 3rd Intl. Conf. 3D Digital Imaging and Modeling. (2001) 145–152
13. Eriksen, R.D.: Image processing library 98 (2006) http://www.mip.sdu.dk/ipl98/.
14. Elgammal, A., Muang, C., Hu, D.: Skin detection - a short tutorial (2009)
15. Deng, Y., Manjunath, B.: Unsupervised segmentation of color-texture regions in images and video. IEEE Trans. PAMI **23**(8) (2001) 800–810
16. Christoudias, C., Georgescu, B., Meer, P.: Synergism in low-level vision. In: Proc. ICPR. Volume 4., Quebec City, Canada (2002) 150–155
17. Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial structures for object recognition. Int. J. Comput. Vision **61**(1) (2005) 55–79
18. Sminchisescu, C., Triggs, B.: Kinematic jump processes for monocular 3D human tracking. In: Computer Vision and Pattern Recognition. (2003) I: 69–76
19. Shotton, J., Blake, A., Cipolla, R.: Efficiently combining contour and texture cues for object recognition. In: British Machine Vision Conference. (2008)
20. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient matching of pictorial structures. In: Proc. ICPR. Volume 2. (2000) 66–73
21. Noriega, P., Bernier, O.: Multicues 2D articulated pose tracking using particle filtering and belief propagation on factor graphs. In: Proc. ICPR. (2007) 57–60
22. Ferrari, V., Marin-Jimenez, M., Zisserman, A.: Progressive search space reduction for human pose estimation. Proc. CVPR (2008) 976–983
23. Niebles, J.C., Han, B., Ferencz, A., Fei-Fei, L.: Extracting moving people from internet videos. In: Proc. ECCV, Berlin, Heidelberg, Springer (2008) 527–540
24. Marcin, E., Vittorio, F.: Better appearance models for pictorial structures. In: Proc. BMVC. (September 2009)
25. Kumar, M.P., Torr, P.H.S., Zisserman, A.: Efficient discriminative learning of parts-based models. Proc. ICCV) (2009)