Unsupervised learning of human body parts from video footage

Thomas Walther and Rolf P. Würtz
Institut für Neuroinformatik, Ruhr-Universität
D-44780 Bochum, Germany
thomas.walther@neuroinformatik.rub.de

Abstract

Estimation of human body postures from video footage is still one of the most challenging tasks in computer vision. Even most recent approaches in this field rely strongly on domain knowledge provided by human supervisors and are nevertheless far from operating reliably under real-world conditions. We propose to overcome these issues by integrating principles of organic computing into the posture estimation cycle, thereby relegating the need for human intervention while simultaneously raising the level of system autonomy.

1. Introduction

Human beings effortlessly analyze and interprete body motion of other individuals (see e. g. [11]). This distinct skill is one of the mainstays of 'social perception' [18], allowing effective and smooth cooperation of human subjects in a complex environment.

Over the last decades, there has been continuous struggle to build artificial *pose estimation* (PE) systems that mimic human skills in retrieving body postures from visual input. Such systems would impact a broad market: applications in health care, surveillance, industry and sports (see e. g. [9], [13], [17]) are obvious.

Yet, despite remarkable research efforts, actual pose estimation systems (see e. g. [21] for a comprehensive overview) are far from rivaling their biological paradigm: while relying on a disproportionate, increasing amount of human supervision, the vast majority of modern PE solutions is unable to operate reliably on real-world scenarios.

In our project, we adress this annoying lack of competetiveness by adopting organic computing [30] paradigms into the PE domain. Our PE approach is designed as to mimic human strategies of unsupervised, 'non-trivial learning' [20]: upper body models combining human appearance and limb kinematics are extracted 'from the input itself' [10], while appropriate generalization strategies guide application of the learned models to new situations.

As demonstrated in e. g. [29], analysis of non-rigid human motion behaviour is feasible: sparse models of the upper human body were learned in an unsupervised manner from well-constrained, simple scenarios.

The contribution of the actual paper is three-fold: first, we propose several methods to increase the performance of the model learning process presented in [29] without sacrificing reliability. Second, the sparse, feature-based body representations derived in [29] are fleshed out using appropriate multi-label image segmentation techniques. Third, the limb templates resulting from the segmentation stage are combined with kinematic constraints learned in [29] to yield a 2D *pictorial structure* model (upper body) of the observed human subjects; generalization potential of the constructed PE system is eventually assessed in scenarios of varying complexity.

2. Enhanced limb proposal extraction

Following the basic limb proposal extraction scheme presented in [29], our solution outperforms the former approach in several aspects; for completeness, we shortly recall the architecture of [29]: in a first step, image features (patches of intensity distributions) are sampled sparsely from given input video footage. These features are subsequently tracked by a differential optical flow scheme [27] through all frames of a given input sequence. Obviously, feature motion reflects human body motion: coherently moving features are likely to represent single limbs. This assumption is exploited by a follow-up 'self-tuning' [16] spectral clustering stage that groups features according to the similarity of their motion trajectories. In a final step, body kinematics are extracted by finding joint connections between the segmented limb clusters. This skeleton extraction mechanism is based on a probabilistic maximum spanning tree (MST) algorithm proposed by [12].

2.1. Guided feature placement

Whereas the above feature sampling strategy sounds straightforward in theory, its practical realization is far from

2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops
978-1-4244-4441-0/09/$25.00 ©2009 IEEE

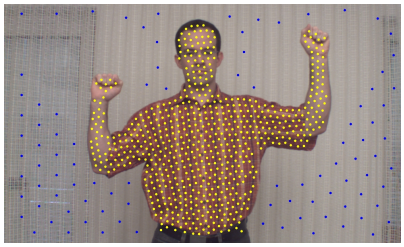


Figure 1: Guided feature placement: foreground features are indicated in yellow, background features are colored blue

trivial: in [29], feature 'selection' had been reduced to mere feature placement, i. e. features were distributed homogeneously on incoming footage. Yet, this strategy yields a tickler: scattering a large number of features wastes a significant amount of processing power on bland background structures. Contrarily, keeping the feature density too low can easily cause tracking loss of important, yet weakly textured, fast-moving body parts (like bare forearms).

To avoid these problems, we employ a feature selection automatism that combines frame differencing strategies, morphological operations and GraphCut [2] mechanisms to concentrate features on the moving human body. Besides lowering computational efforts and rendering feature tracking more reliable, this practice provides foreground/background labelings for each feature. Fig. 1 depicts exemplary results of the enhanced feature placement scheme.

2.2. Revisiting spectral clustering

In [29] a complex heuristic functional was proposed to guide the spectral clustering stage. Here, we were able to reduce the heuristic complexity while keeping up segmentation quality by introducing a fully automatic post-processing stage: spectral clustering now works according to the well-established, comprehensive 'normalized cut' criterion proposed by [23]. This baseline technique generates a segmentation structure that reflects the true body part configuration already quite well: let each feature cluster in the generated segmentation be henceforth termed a *limb fragment* .

However, the achieved segmentation is not perfect: excess 'rogue' fragments might be detected due to cloth stretching or joint activity as depicted in fig. 2a. By employing a fully autonomous merge/split scheme in a separate post-processing stage, these rogue fragments are eliminated: during the merging phase, all rogue fragments which approximately keep their relative rotation (± 15 degrees) in all frames of the sequence are fused; this simple heuristic effectively counters rogues induced by cloth motion.

Treating rogue fragments resulting from joint activity requires a different approach: each potential joint rogue fragment (each fragment having exactly two neighbor fragments) is putatively split and divided amongst its neighboring fragments; after each split, the skeletal tree is reconstructed. The split that optimally preserves skeletal quality (measured heuristically as the sum of logarithmic edge plausibilities in the skeletal MST, w. r. t. the initial fragment configuration) is then put into practice and the process iterates, starting from the new fragment configuration. This greedy scheme is stopped if elimination of any potential joint rogue would significantly deteriorate the overall skeletal quality.

The effectiveness of our rogue annihilation solution is depicted in fig. 2b.

2.3. Single-shot sequence analysis

Adding up to the above enhancements, performing expensive spectral clustering in each input frame (as suggested in [29]) turned out to be unnecessary: since the last frame of an input sequence integrates motion information from all previous frames, limiting spectral clustering efforts to this last frame does not degrade overall segmentation results. Yet, system performance is significantly boosted: whereas sparse limb learning in [29] took ≈ 10 hrs. for a standard input sequence, our strategy solves the problem in a couple of minutes.

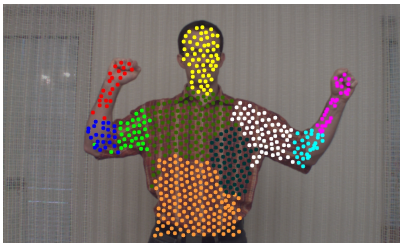
3. Limb refinement

The constructed sparse limb proposals give only a very approximate idea of true human body (limb) shape. To set up a full-fledged 2D model of the upper human body, fleshing out the body part approximations becomes mandatory.

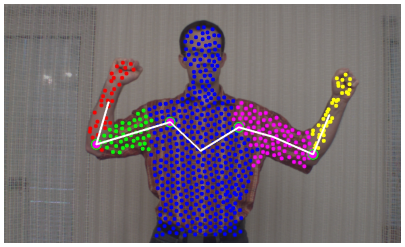
This problem essentially turns out as a multi-label image segmentation task: based on information stemming from the K_P sparse limb proposals which passed the rogue fragment elimination stage, all unclassified image pixels have to be assigned to a dedicated body part; to simplify further discussion, the background is treated as an additional 'limb' in the following.

Binary (two-label) image segmentation problems can globally be solved from within a fast graph-cut-based energy minimization framework [2]; Veksler ([28]) extends this graph cut segmentation concept to multiple labels: her α*-expansion* and $\alpha\beta$*-swap* algorithms internally bank on binary graph cut mechanisms [25] to find quickly a 'strong local minimum' of the selected, multi-label objective (energy) function.

To apply Veksler's ideas to the limb segmentation problem (as done in similar form by [15]), appropriate set-up of the mentioned energy function is essential; to begin with, assume that segmentation is performed in a dedicated frame at time t_0 (henceforth, this frame is also called *reference*



(a) Fragments as received from the spectral clustering stage



(b) Fragment configuration after post-processing; cloth and joint rogues are annihilated

Figure 2: Rogue cluster removal by heuristic post-processing, skeleton overlaid

frame), let $\mathcal{X} = \{\mathbf{x}_0, ..., \mathbf{x}_{N_P}\}$ identify all pixels in the processed frame. $\mathbf{S}$ be a *multi-label segmentation vector*, with components $S_i = l$ if pixel $\mathbf{x}_i$ is currently assigned to limb l. As we operate on full color images, let vector $\mathbf{I}(\mathbf{x}_i, t)$ paraphrase the RGB-value of pixel $\mathbf{x}_i$ at time t.

Then, the energy $E(\mathbf{S})$ corresponding to a dedicated configuration of the segmentation vector (such a single configuration of $\mathbf{S}$ is called a 'labeling' of the image pixels) can appropriately be formulated as follows (adopted from [15] and [1]):

$$E(\mathbf{S}) = \lambda \underbrace{\sum_{\mathbf{x}_i \in \mathcal{X}} R_{\mathbf{x}_i}(S_i)}_{E_O} + \underbrace{\sum_{(\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{N}} B_{(\mathbf{x}_i, \mathbf{x}_j)} \cdot (1 - \delta(S_i, S_j))}_{E_B} \quad (1)$$

with

$$\delta(S_i, S_j) = \begin{cases} 1 & \text{if } S_i = S_j \\ 0 & \text{otherwise} \end{cases}$$

and

$$B_{(\mathbf{x}_i, \mathbf{x}_j)} \propto e^{-\frac{(I(\mathbf{x}_i, t_0) - I(\mathbf{x}_j, t_0))^2}{2\sigma^2}} \cdot \frac{1}{\|\mathbf{x}_i - \mathbf{x}_j\|}$$

$\mathcal{N}$ here identifies the complete set of pixel pairs defined under a standard 4-neighborhood system (see e. g. [1]). Eventually, note that $\lambda = 0.01$ and $\sigma = 10.0$ in all our experiments.

Interpretation of eq. 1 is straightforward: the observation energy term E_O takes on minimum values iff a current labeling $\mathbf{S}$ complies with model knowledge gained from the limb proposals; thus, $R_{\mathbf{x}_i}(l)$ describes how well observations stemming from a certain reference frame pixel $\mathbf{x}_i$ can be explained by the l^{th} limb model. The boundary term E_B is minimized iff $\mathbf{S}$ comprises spatially extended, coherent limb templates; the $B_{(\mathbf{x}_i, \mathbf{x}_j)}$ term additionally constrains inter-limb boundaries to coincide with natural image intensity boundaries.

To further quantify $R_{\mathbf{x}_i}$, we draw inspiration from [12], exploiting the *motion* , *color* and *shape* cues each limb proposal provides.

Without any further domain knowledge, it is convenient to assume the RGB color distribution of each limb to be approximately Gaussian [12]; thus

$$M_l^C(\mathbf{x}_i) = \frac{1}{\sqrt{(2\pi)^3 \left|\Sigma_l^C\right|}} \cdot e^{-\frac{1}{2}\left(I(\mathbf{x}_i, t_0) - \mu_l^C\right)^T \left(\Sigma_l^C\right)^{-1} \left(I(\mathbf{x}_i, t_0) - \mu_l^C\right)} \quad (2)$$

constitutes an appropriate limb *color model* for each body part l; the RGB-mean μ_l^C and the full covariance matrix Σ_l^C are learned by ML estimation from the corresponding, sparse limb proposals.

As well, the spatial distribution of all pixels assigned to each limb is assumed to be of Gaussian type [12], yielding the limb *shape model*

$$M_l^S(\mathbf{x}_i) = \frac{1}{\sqrt{(2\pi)^3 \left|\Sigma_l^S\right|}} e^{-\frac{1}{2}\left(\mathbf{x}_i - \mu_l^S\right)^T \left(\Sigma_l^S\right)^{-1} \left(\mathbf{x}_i - \mu_l^S\right)} \quad (3)$$

where μ_l^S and $\mathbf{\Sigma}_l^S$ are again derived from the limb proposals by ML estimation.

Eventually, observe that pixels moving coherently with the l^{th} limb proposal are likely to belong to the l^{th} body part. The degree of motion coherence between a single reference frame pixel and a certain limb l is readily assessed: given the positional and rotational changes that limb proposal l undergoes when transiting from time t to time t_0, a transformation $T_l^{t \to t_0}(\mathbf{x})$ can be set up which maps pixels from any frame at time t to the reference frame, inherently hypothesizing that each transformed pixel moves coherently with body part l. It should be immediately clear that this warping process causes the difference $\left|I(\mathbf{x}_i, t_0) - I(T_l^{t \to t_0}(\mathbf{x}_i), t)\right|$ to become small only if the above hypothesis holds and the projected pixel $\mathbf{x}_i$ is truly part of the l^{th} limb. This insight

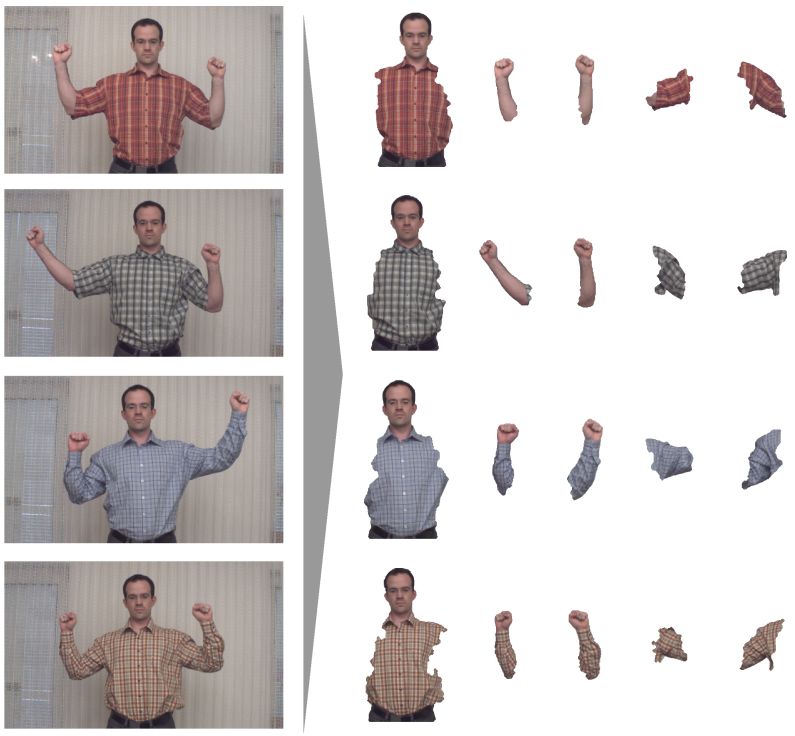


Figure 3: Limb templates automatically extracted from some input sequences

allows to establish a *forward motion model* [12]

$$M_l^{M+}(\mathbf{x}_i) = \prod_{t=t_0+1}^{t_0+R^+} \frac{1}{\sqrt{2\pi\sigma_M^2}} e^{-\frac{1}{2}\frac{\left|I(\mathbf{x}_i,t_0)-I(T_l^{t\to t_0}(\mathbf{x}_i),t)\right|^2}{\sigma_M^2}} \tag{4}$$

that is based on image intensity information from R^+ frames following the reference frame; note that σ_M is set to 10.0 in all our experiments; $R+$ is set to a low value (3) to keep motion blur at bay.

By plugging the above partial models together, a *combined limb model*

$$M_l(\mathbf{x}_i) = M_l^C(\mathbf{x}_i) \cdot M_l^S(\mathbf{x}_i) \cdot M^{M+}(\mathbf{x}_i) \tag{5}$$

can be formulated, whose negative logarithm plays the role of the sought-after observation cost function in eq. 1, such that

$$R_{\mathbf{x}_i}(S_i) = -ln(M_{S_i}(\mathbf{x}_i)) \tag{6}$$

With that, the above energy functional is fully defined and Veksler's alpha expansion algorithm (for details concerning this algorithm, see e. g. [28], [25]) can be lauched on the reference frame data. To cancel out potential motion blur caused by temporal integration in eq. 4, we follow [12] and repeat the above segmentation process after replacing the forward motion model in eq. 5 with a similar *backward motion model*

$$M^{M-}(\mathbf{x}_i) = \prod_{t=t_0-1}^{t_0-R^+} \frac{1}{\sqrt{2\pi\sigma_M^2}} e^{-\frac{1}{2}\frac{\left|I(\mathbf{x}_i,t_0)-I(T_l^{t\to t_0}(\mathbf{x}_i),t)\right|^2}{\sigma_M^2}} \tag{7}$$

that integrates information from R^- frames preceeding the reference frame; for reasons of symmetry, $R^- = R^+ = 3$.

Binarized forward/backward segmentation masks for each limb (derived from the forward/backward segmentation vectors) are then merged through a logic 'and', thereby reliably eliminating motion blur.

Finally, morphological opening and closing operators are applied to the resulting *limb templates*, removing minor segmentation artifacts and annealing small hiatuses found in the extracted limb layers, while leaving the overall limb shape largely unaltered.

Limb templates extracted from our video footage are shown exemplary in fig. 3. While the above techniques yield acceptable results in our trials on simple backgrounds, future implementations offer chances for improvements: esp. the unimodal Gaussian used to define the limb color models is quite restrictive and could be replaced by a Gaussian mixture model. This strategy would allow to employ iterative, GrabCut-like [22] methods for color model adaptation, potentially increasing segmentation precision in more complex scenarios with stronger background clutter.

4. Pose estimation

In general, PE systems can be classified according to many criteria, see [9] or [21]; the system we propose is most closely related to 2D bottom-up solutions (like e. g. [24]) and baseline techniques from this domain are integrated in our approach, allowing for autonomous pose inference from still images or single frames of a video sequence. However, the body model needed in standard bottom-up pose finding is generally provided by a human supervisor; this manual intervention clearly abates systemic self-reliance.

We completely abandon human supervision from the process loop: the sought-after model of the upper human body is generated automatically by combining the extracted kinematic skeleton with the learned limb layers.

4.1. The human body model

In bottom-up PE approaches, a 2D body model combining limb appearance information and skeletal kinematic constraints is often formulated as a 'pictorial structure' (PS) [7] model graph: nodes in the graph represent the single limbs, whereas the graph edges enforce kinematic relationships between the body parts.

Appearance cues encoded in the extracted limb templates span cloth color, texture, illumination conditions and limb shape. Yet, PS body models built on color, texture or illumination become highly subject- and/or scenario-specific, rendering their generalization capabilities negligible. Contrarily, prohibiting excessive limb foreshortening and ensuring stable subject-camera distances, limb shapes can be expected to remain rather invariant with respect to the captured (adult) subject and environmental conditions.

As a consequence of these insights, we base our PS model on limb shapes, temporarily discarding other sources of information. Note, however, that the color information is not lost, but stored for later use. The resulting shape-based PS model displays good generalization capabilities in the experiments described below.

4.2. Inferring 2D human body poses

Pictorial structures naturally unify apperance and kinematic constraints of a modeled subject/object: in the current context, the employed upper body model comprises a tree-like graph $G(\mathcal{V}, \mathcal{E})$, with its vertices $\mathcal{V} = \{\mathbf{v_0}, ..., \mathbf{v}_{K_P}\}$ representing shape of the K_P single body part templates that result from limb refinement. The vertices are connected by edges $\mathcal{E} = \{e_0, ...\}$ which encode pairwise kinematic relations (joint constraints from the skeletal MST) between the identified parts. Loosely, tree edges can be imagined as 'springs', keeping together the limbs at the body joint positions [6]. Edge weights are employed to encode stiffness of the springs and inherently control model flexibility.

Assume that each body part i stored in the model tree obeys a rigid motion model, allowing for translation (x_i, y_i), rotation (θ_i) and scaling (s_i); let a dedicated point in the corresponding limb state space be henceforth termed a *location* [6] $\mathbf{l}_i$ of the considered body part. The set $\mathcal{L} = \{\mathbf{l}_0, ...\mathbf{l}_{K_P}\}$ of locations for all body parts is henceforth termed a model *configuration* [6]. Each location is allowed to take on only discrete values; a fine discretization grid is employed to mimic the continuous nature of limb parameter spaces (an idea borrowed from [8]).

Given a single input frame of a video sequence (or a single still image) with image observations $\mathbf{I}$, let $m_i(\mathbf{l}_i, \mathbf{I})$ be a '*match cost function* that measures how well body part i matches the image observations $\mathbf{I}$ when placed at location $\mathbf{l}_i$' (loosely adopted from [6]).

As we base our PS model on limb shape, calculation of $m_i(\mathbf{l}_i, \mathbf{I})$ naturally builds on image edges. To construct appropriate observations $\mathbf{I}$, the input image has to be converted into an edge map; necessary edge extraction could be done by fast, gradient-based schemes (like Canny detection [3]). However, the results produced by such basic edge detectors often depend on careful, manual finetuning of systemic parameters (which conflicts with OC principles). Furthermore, excess edges (e. g. in textured image areas) constitute a severe problem in baseline edge detection and can significantly deteriorate the reliability of the limb matching process. For this reason, we employ the more sophisticated JSEG [5] edge extraction scheme that combines color and texture cues to find salient image edges. This practice effectively reduces excess edges and the need for manual intervention. Thus, $\mathbf{I}$ describes the edge observations derived from the input image by applying the JSEG algorithm. Note, for completeness, that other promising edge extraction schemes exist (e. g. [4]) which could also be employed for edge analysis in the current context.

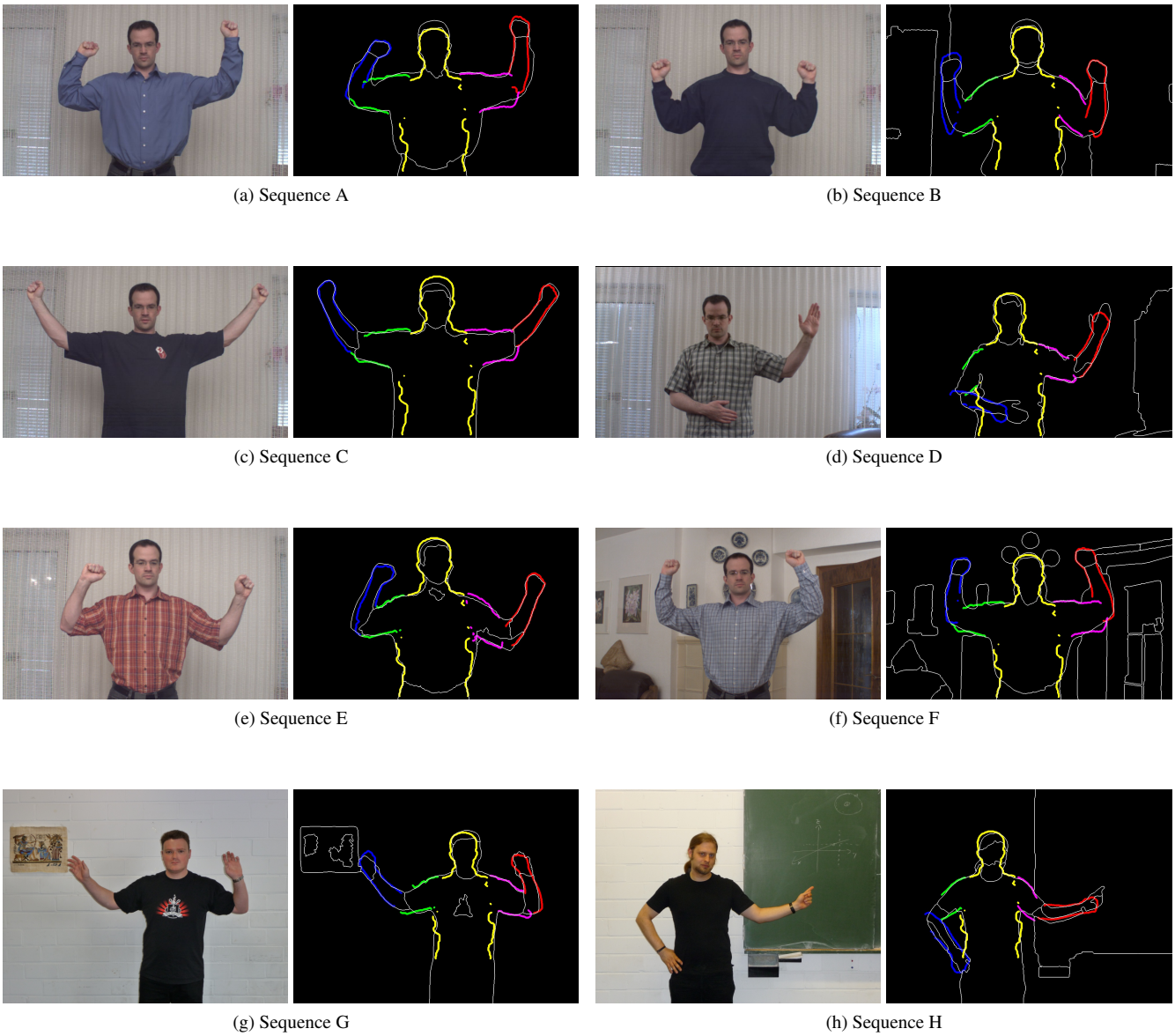
The limb templates are then matched to a *chamfer distance* image (cf. e. g. [26]) derived from these edge observations; this widespread practice yields good matching results in general.

In addition to the above matching cost term, set up a *deformation cost function* [6] $d_{ij}(\mathbf{l}_i, \mathbf{l}_j)$; this function evaluates graph edge information and takes on low values iff placing body parts i and j at locations $\mathbf{l}_i, \mathbf{l}_j$ does not violate any kinematic model constraints; in other words, $d_{ij}(\mathbf{l}_i, \mathbf{l}_j)$ penalizes model configurations that do not comply with valid human body assemblies (w. r. t. the learned kinematic skeleton).

Putting the above cost functions together yields the energy/cost functional [7]

$$E^{PS}(\mathcal{L}) = \left(\sum_{\mathbf{v}_i \in \mathcal{V}} m_i(\mathbf{l}_i, \mathbf{I}) + \sum_{(\mathbf{v}_i, \mathbf{v}_j) \in \mathcal{E}} d_{ij}(\mathbf{l}_i, \mathbf{l}_j) \right) \quad (8)$$

that sums up the total cost for matching a dedicated config-



(a) Sequence A (b) Sequence B

(c) Sequence C (d) Sequence D

(e) Sequence E (f) Sequence F

(g) Sequence G (h) Sequence H

Figure 4: Analyzing novel sequences: original images on the left, JSEG edge images on the right. The best matching result is overlaid to the JSEG images.

uration of the given PS model to the provided image observations $\mathbf{I}$.

It is readily seen [6] that a model configuration

$$\mathcal{L}^* = \arg\min_{\mathcal{L}} E^{PS}(\mathcal{L}) \tag{9}$$

is the globally optimal solution to the pose inference problem at hand.

Due to the enormous complexity of the given minimization problem, finding a globally optimal solution is far from trivial: for arbitrary graphs, algorithm runtime becomes $O(m^n)$, with m being the number of allowed discrete locations per limb and n representing the number of graph vertices [6]. As m typically grows large [6], an exact solution of the general pose inference problem quickly becomes unwieldy on availiable computer hardware. Yet, in the current context, there is no need to handle arbitrary model graphs, as we constrain the kinematic skeleton to be tree-structured. For such simplified, tree-like graphs, [6] proposes to find $\mathcal{L}^*$ in $O(mn)$ time by employing dynamic programming techniques; we follow this approach and use a variant of the

Viterbi algorithm to perform fast upper-body pose inference in all captured scenarios. For a detailed description of the applied Viterbi method, [6] and [7] are recommended.

5. Assessing generalization capabilities

Using the techniques described above, it remains to analyze to what degree the resulting PE system is able to generalize, i. e. to 'apply what has been learned from limited experience to new situations' [20]. We assess generalization capabilities by first learning a single PS model of the upper human body from one of our training sequences. During the following experiments, the learned model is overlaid to the corresponding JSEG edge images; individual body parts are marked in different colors.

The generated model is then used for pose estimation in all residual testing scenarios in fig. 4: from fig. 4a - fig. 4e it can be deduced that the system generalizes well over the type of worn apparel, cloth color and texture. Furthermore, changing motion patterns and soft cloth deformation (fig. 4b) are coped with quite satisfactorily. Note one matching subtlety in fig. 4d and 4h: though the right forearm is matched to the correct position, the limb appears contorted. As our system currently does not probe out-of-plane rotations of the body parts, such behavior is expected; future system versions will remedy this by allowing for limb flipping in addition to pure 2D rotation. Fig. 4f eventually demonstrates matching capabilities in scenarios with increased background clutter.

Eventually, as demonstrated in fig. 4g and fig. 4h, the matching scheme generalizes acceptably across different individuals, given changing environmental conditions.

6. Conclusions and future perspectives

We proposed an artificial PE system that overcomes entrapments of conventional PE solutions by integrating principles of organic computing: the system learns models of the upper human body w. r. t. limb appearance and kinematics without human supervision. The learned model is successfully applied to retrieve body postures in novel input footage.

Compared to other systems that perfom limb learning from video footage, our solution has several advantages: [12], though acting as a basis for our approach, displays several issues w. r. t. system autonomy. Krahnstoever relies on human intervention to purport the correct number of retrieved limbs and learns scenario-specific 3D models that display minor generalization capabilities. In contrast, the number of limbs is found automatically in our solution (as a side-effect of post-processing), while the learned 2D body models are kept quite generic, yielding, in combination with bottom-up techniques, acceptable generalization performance.

A remarkable, layer-based limb extraction scheme has recently been proposed by [15]. Though showing promising segmentation results for quite generic scenarios, the role of non-rigidity is not explicitly explored yet, furthermore, the presented system structure becomes quite complex. On the other hand, our system handles moderate non-rigidity efficiently in the heuristic post-processing stage, while keeping the system architecture complexity low.

To be understood as a possible extension to [15], the bottom-up model matching method found in [14] works on complete model graphs and attaches no importance to finding a 'correct', tree-like kinematic skeleton. While this strategy is sufficient for pure detection of articulated entities, our approach steps further and aims at exact posture retrieval of articulated human bodies.

Still, several issues have to be adressed in future research: currently, generalization had been tested for one training sequence and a variety of test sequences. It will be interesting to see, in how far generalization capabilities withstand changes of the model training sequence (e. g. using a shirt with long sleeves for model training).

Furthermore, the model used so far is quite oversimplified; besides limb shape, hand and face color distribution could be exploited as another rather invariant cue, making limb matching more reliable (disregarding camouflage attempts). The question to be answered by follow-up research is how to learn such pertinent color distributions automatically from a multitude of extracted models. Similarily, by combining multiple models, it should be possible to learn an average *prototype shape* for each observed body part. Such prototypical shapes are likely to provide a good tradeoff between preservation of important shape characteristics and limb matching performance. Eventually, limb symmetries (e. g. both forearms display a similar color distribution) might be exploited to render matching more robust.

From the model registration point of view, two enhancements are planned: first, it is obvious that some body parts (torso, forearms) can be matched more reliably to new image content than other limbs. Learning such limb saliency automatically and exploiting it during model matching shall be explored in future research. Furthermore, joint limits can be inferred from given input footage; knowing these limits allows to prevent pose estimation to return implausible body poses.

Finally, replacing the simple Viterbi algorithm with more complex Belief Propagation (cf. e. g. [31]) schemes would allow for models with loops (necessary for occlusion handling and intersection prevention, see [24] and [19]). Another future option is to integrate more complex scenario analysis strategies (e. g. [15]) into our own approach, thereby allowing for model learning from less restricted situations.

Acknowledgments

Funding by the Deutsche Forschungsgemeinschaft (MA 697/5-1, WU 314/5-2) is gratefully acknowledged.

References

[1] Y. Boykov and G. Funka-Lea. Graph cuts and efficient n-d image segmentation. *International Journal of Computer Vision*, 70(2):109–131, 2006.

[2] Y. Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In *Processdings of the International Conference on Computer Vision*, volume 1, pages 105–112, Vancouver, Canada, 2001.

[3] J. F. Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence*, 8:679–698, 1986.

[4] J. Chen, T. N. Pappas, A. Mojsilovic, and B. E. Rogowitz. Adaptive perceptual color-texture image segmentation. *IEEE Transactions on Image Processing*, 14(8):1524–1536, 2005.

[5] Y. Deng and B. S. Manjunath. Unsupervised segmentation of color-texture regions in images and video. *Pattern Analysis and Machine Intelligence*, 23(8):800–810, 2001.

[6] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient matching of pictorial structures. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, page 2066, 2000.

[7] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *Int. J. Comput. Vision*, 61(1):55–79, 2005.

[8] V. Ferrari, M. Marin, and A. Zisserman. Progressive search space reduction for human pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[9] D. M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–98, 1999.

[10] P. F. Gorder. Neural networks show new promise for machine vision. *Computing in Sciene and Engineering*, pages 4–8, 2006.

[11] G. Johannson. Visual perception of biological motion and a model for its analysis. *Perception and Psychophysics*, 14(2):201–211, 1973.

[12] N. Krahnstoever. *Articulated Models from Video*. PhD thesis, Pennsylvania State University, 2003.

[13] N. Krahnstoever, M. Yeasin, and R. Sharma. Automatic acquisition and initialization of articulated models. *Machine Vision and Applications*, 14:218–228, 2003.

[14] M. P. Kumar, P. H. S. Torr, and A. Zisserman. Extending pictorial structures for object recognition. In *Proceedings of the British Machine Vision Conference*, pages 789–798, 2004.

[15] M. P. Kumar, P. H. S. Torr, and A. Zisserman. Learning layered motion segmentations of video. *International Journal of Computer Vision*, 76(3):301–319, 2008.

[16] L. Z. Manor and P. Perona. Self-tuning spectral clustering. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 1601–1608. MIT Press, 2005.

[17] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104:90–126, 2006.

[18] J. P. Morris, K. A. Pelphrey, and G. McCarthy. Occipitotemporal activation evoked by the perception of human bodies is modulated by the presence or absence of the face. *Neuropsychologia*, 44:1919–1927, 2006.

[19] P. Noriega and O. Bernier. Multicues 2d articulated pose tracking using particle filtering and belief propagation on factor graphs. In *International Conference on Image Processing*, volume 5, pages 57–60, 2007.

[20] T. Poggio and E. Bizzi. Generalization in vision and motor control. *Nature*, 431:768–774, 2004.

[21] R. Poppe. Vision-based human motion analysis: An overview. *Computer Vision and Image Understanding*, 108:4–18, 2007.

[22] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23(3):309–314, 2004.

[23] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[24] L. Sigal and M. J. Black. Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2041–2048, New York, NY, June 2006.

[25] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):1068–1080, 2008.

[26] A. Thayananthan, B. Stenger, P. H. S. Torr, and R. Cipolla. Shape context and chamfer matching in cluttered scenes. *Computer Vision and Pattern Recognition*, 1:127–133, 2003.

[27] C. Tomasi and T. Kanade. Shape and motion from image streams: a factorization method - detection and tracking of point features. Technical report, Carnegie Mellon University, 1991.

[28] O. Veksler. *Efficient Graph-Based Energy Minimization Methods in Computer Vision*. PhD thesis, Cornell University, 1999.

[29] T. Walther and R. P. Würtz. Learning to look at humans - what are the parts of a moving body. In *Lecture Notes in Computer Science*, pages 22–31. Springer, 2008.

[30] R. P. Würtz, editor. *Organic Computing*. Springer Verlag, 2008.

[31] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*, pages 239–269. Morgan Kaufmann Publishers Inc., 2003.