UNIVERSITÄT AUGSBURG
Universitätsbibliothek

OPEN ACCESS
OPUS AUGSBURG

Automatic pose initialization of swimmers in videos

Christian X. Ries, Rainer Lienhart

Angaben zur Veröffentlichung / Publication details:

Ries, Christian X., and Rainer Lienhart. 2010. "Automatic pose initialization of swimmers in videos." In *Visual Information Processing and Communication: IS&T / SPIE Electronic Imaging, 17-21 January 2010, San Jose, California, USA*, edited by Amir Said and Onur G. Guleryuz, 75430J. Bellingham, WA: SPIE. https://doi.org/10.1117/12.838727.

Nutzungsbedingungen / Terms of use:

licgercopyright

Dieses Dokument wird unter folgenden Bedingungen zur Verfügung gestellt: / This document is made available under these conditions:
Deutsches Urheberrecht
Weitere Informationen finden Sie unter: / For more information see:
https://www.uni-augsburg.de/de/organisation/bibliothek/publizieren-zitieren-archivieren/publiz/

PROCEEDINGS
IS&T / SPIE
Electronic
Imaging
SCIENCE AND TECHNOLOGY

Visual Information Processing and Communication

Amir Said
Onur G. Guleryuz
Editors

19–21 January 2010
San Jose, California, United States

Sponsored and Published by
IS&T—The Society for Imaging Science and Technology
SPIE

Volume 7543

Proceedings of SPIE, 0277-786X, v. 7543

The papers included in this volume were part of the technical conference cited on the cover and title page. Papers were selected and subject to review by the editors and conference program committee. Some conference presentations may not be available for publication. The papers published in these proceedings reflect the work and thoughts of the authors and are published herein as submitted. The publishers are not responsible for the validity of the information or for any outcomes resulting from reliance thereon.

Please use the following format to cite material from this book:
Author(s), "Title of Paper," in *Visual Information Processing and Communication*, edited by Amir Said, Onur G. Guleryuz, Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 7543, Article CID Number (2010).

ISSN 0277-786X
ISBN 9780819479365

Copublished by
SPIE
P.O. Box 10, Bellingham, Washington 98227-0010 USA
Telephone +1 360 676 3290 (Pacific Time) · Fax +1 360 647 1445
SPIE.org
and
IS&T—The Society for Imaging Science and Technology
7003 Kilworth Lane, Springfield, Virginia, 22151 USA
Telephone +1 703 642 9090 (Eastern Time) · Fax +1 703 642 9094
imaging.org

Copyright © 2010, Society of Photo-Optical Instrumentation Engineers and The Society for Imaging Science and Technology.

Copying of material in this book for internal or personal use, or for the internal or personal use of specific clients, beyond the fair use provisions granted by the U.S. Copyright Law is authorized by the publishers subject to payment of copying fees. The Transactional Reporting Service base fee for this volume is $18.00 per article (or portion thereof), which should be paid directly to the Copyright Clearance Center (CCC), 222 Rosewood Drive, Danvers, MA 01923. Payment may also be made electronically through CCC Online at copyright.com. Other copying for republication, resale, advertising or promotion, or any form of systematic or multiple reproduction of any material in this book is prohibited except with permission in writing from the publisher. The CCC fee code is 0277-786X/10/$18.00.

Printed in the United States of America.

Paper Numbering: Proceedings of SPIE follow an e-First publication model, with papers published first online and then in print and on CD-ROM. Papers are published as they are submitted and meet publication criteria. A unique, consistent, permanent citation identifier (CID) number is assigned to each article at the time of the first publication. Utilization of CIDs allows articles to be fully citable as soon they are published online, and connects the same identifier to all online, print, and electronic versions of the publication. SPIE uses a six-digit CID article numbering system in which:

- The first four digits correspond to the SPIE volume number.
- The last two digits indicate publication order within the volume using a Base 36 numbering system employing both numerals and letters. These two-number sets start with 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B ... 0Z, followed by 10-1Z, 20-2Z, etc.

The CID number appears on each page of the manuscript. The complete citation is used on the first page, and an abbreviated version on subsequent pages. Numbers in the index correspond to the last two digits of the six-digit CID number.

Automatic Pose Initialization of Swimmers in Videos

Christian X. Ries and Rainer Lienhart

Multimedia Computing Lab, University of Augsburg, Augsburg, Germany
{ries,lienhart}@informatik.uni-augsburg.de

ABSTRACT

We propose an approach to the task of automatic pose initialization of swimmers in videos. Thus, our goal is to detect a swimmer inside a target video and assign an estimated position to her/his body parts. We first apply a non-skin-color filter to reduce the search space inside each target frame. We then match previously devised template sequences of Gaussian feature descriptors against sequences of feature vectors which are computed within the remaining image regions. Finally, relative average joint positions from annotated images featuring the key pose are assigned to the detection result and three-dimensional joint positions are estimated. We present detection results for test videos of three different swim strokes and examine the performance of four types of feature descriptors.

1. PROBLEM STATEMENT AND MOTIVATION

For the past decade the analysis of human motion has been a heavily investigated subject by the computer vision community. Especially the analysis of sports videos has gained a remarkable amount of attention in computer vision and sports science research in recent years. One major goal is to automatically evaluate quantitatively an athlete's performance under training and racing conditions continuously in order to help coaches to adjust the training schedule to an athlete's personal needs as well as to analyze a competitor's edge. A first key problem of continuous pose analysis is to find the location of an athlete inside of a video and to identify reliably her or his pose. This first key problem of pose analysis, usually referred to by *pose initialization*, is the subject of this paper. It commonly serves as the starting point for any subsequent pose tracking through time in a video.

In this paper we concentrate on videos of swimmers for several reasons: On the one hand motion analysis is especially useful for individual sports such as swimming where the athlete's success vitally depends on his actual movement patterns. On the other hand swimming videos feature two additional challenges compared to videos of most other sports: (1) An athlete is usually not fully visible from pool side cameras as it is hard to see below the water surface and (2) the background, which is the water surface of the swimming pool, is highly noisy, wavy and specular. Thus, in our framework we first identify and exclude background areas, before using feature descriptors to model the visible part of a swimmer's body from a few template images. In general we can expect that a technique working under these difficult conditions should also be applicable to other visually easier observable individual sports such as running or long jump.

2. RELATED WORK

As mentioned in the introduction there are many approaches to human pose analysis and its closely related disciplines such as pose tracking, which generally require pose initialization as a first major step. Commonly a human model or a mixture of several human models is utilized for recognizing poses. Examples are a stick figure model consisting of the main human joints or a body contour model. The latter often also requires an underlying stick figure model to deduce constraints for human body postures.

For instance, stick figure models were used by Cheng et al.[4] to find possible human postures by applying geometrical projection theory to video frames. Baharatkumar et al.[5] used 3D kinematic data of legs of walking people which they projected onto two-dimensional images. Leung et al.[6] used a model of structural relations and shape relations between body parts to label two-dimensional contours of human body postures in images. Ren et al.[7] also used such relations for pose estimation. Their constraints were postulated for pairs of body parts, which were identified by finding parallel lines inside a discrete graph of contours. Mori et al.[8] tried to recognize human poses by transforming example query contours into contours found in target images and

Visual Information Processing and Communication, edited by Amir Said, Onur G. Guleryuz,
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 7543, 75430J · © 2010 SPIE-IS&T
CCC code: 0277-786X/10/$18 · doi: 10.1117/12.838727

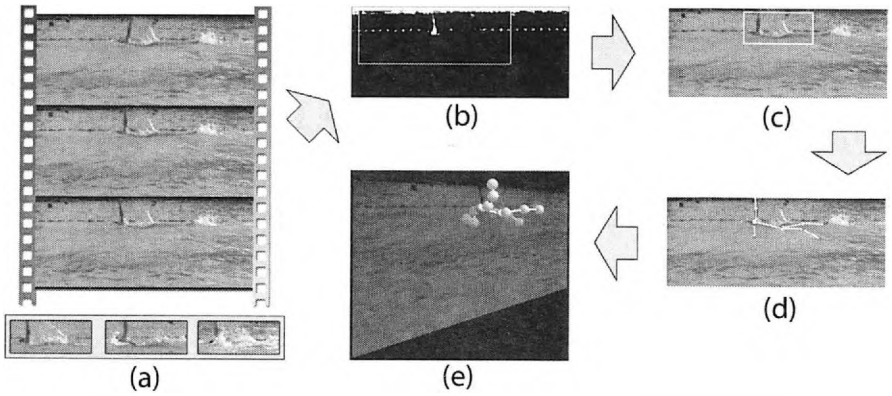

Figure 1. Pose initialization framework: (a) INPUT: A backstroke swimming video (top) and a few template images of a key backstroke pose (bottom) (b) Candidate swimmer areas inside each frame identified by applying the hue filter (white pixels show skin-colored pixels) (c) Region which most likely includes a swimmer in the pose of one of the template images using Gaussian matching with self-similarity descriptors (d) Assigned 2D joint positions (e) OUTPUT: Estimated three-dimensional pose.

evaluating a cost function in doing so. Agarwal et al.[9] used shapes encoded by histograms of shape contexts and trained various regressors on these histograms. Shape context descriptors were also used by Andriluka et al.[10] in order to detect people and estimate their articulated pose in videos by introducing a generic probabilistic model. Spatial-temporal shape templates can be created from motion capture data as suggested by Dimitrijevic et al.[11] who then matched these templates against silhouettes in image sequences. Rogez et al.[12] proposed randomized trees trained on histograms of oriented gradients (HOG) for human pose recognition from videos.

All of these approaches however presume multiple human limbs to be visible in each frame, which is usually not the case for swimming videos. Hence, we cannot apply these respective models unmodified. Furthermore most approaches require a large database of training examples or rely on motion capture data, neither of which was available for our videos. Therefore we concentrate on one characteristic pose per swimming stroke, which the swimmer will definitely take at some point in time during each stroke cycle. Our approach to automatically finding this pose in swimming videos includes color analysis in the HSV color space as a prefilter, which was similarly used by Tong[13] as well as exploiting integral images for speed-up (see Viola[14] and Lienhart[15]). The subsequent swimmer detection is based on the approach of Shechtman[16] who also introduced the self-similarity descriptors. The underlying probabilistic matching approach was suggested by Moghaddam et al.[17] Furthermore, we compare the performance of the self-similarity features to several alternative feature descriptors such as SIFT,[18] Geometric Blur[19] and HOG.[20] Finally, three-dimensional joint positions are estimated by applying the method of Taylor.[21] This work was also inspired by the works of Tong et al.[22] and Liao et al.[23] who analyzed swimming videos to detect the periodicity of local motion and then to deduce the swimming style.

3. POSE INITIALIZATION FRAMEWORK

Our goal is to find a swimmer in a predetermined characteristic key pose of a swim stroke as accurately as possible by searching the video clip for the frame with the best match. After finding the swimmer's location in a video frame where he or she takes this key pose, we assign the relative positions of the swimmer's joints in the template to the actual frame found in order to estimate the 3D pose of the swimmer.

Figure 1 gives an overview of the various steps we designed to attack the problem of pose initialization of swimmers. A swim video of swimmers swimming a specific swim style (Figure 1(a) top) and template images collected manually in advanced showing swimmers taking a key pose of the swim style under detection (Figure 1(a) bottom) serve as the input to our framework. Figure 2 shows some enlarged examples of template images of key poses for three different swimming styles.

The pose estimation is performed in three steps starting (1) with a fast reduction of the search space by applying a skin color model to each video frame to identify candidate locations of swimmers (see Figure 1(b)). Then the average ratio and the standard deviation of skin-colored pixels to non-skin-colored pixels for our

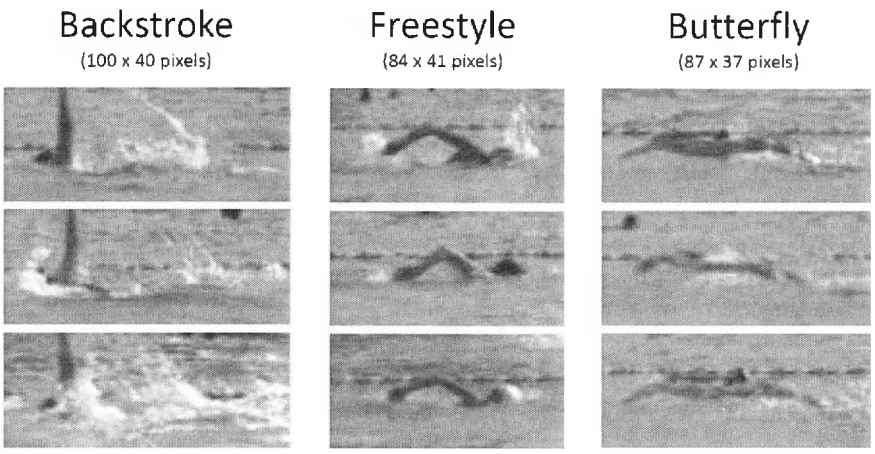

Figure 2. Template images of key poses for the swimming strokes backstroke, freestyle, and butterfly.

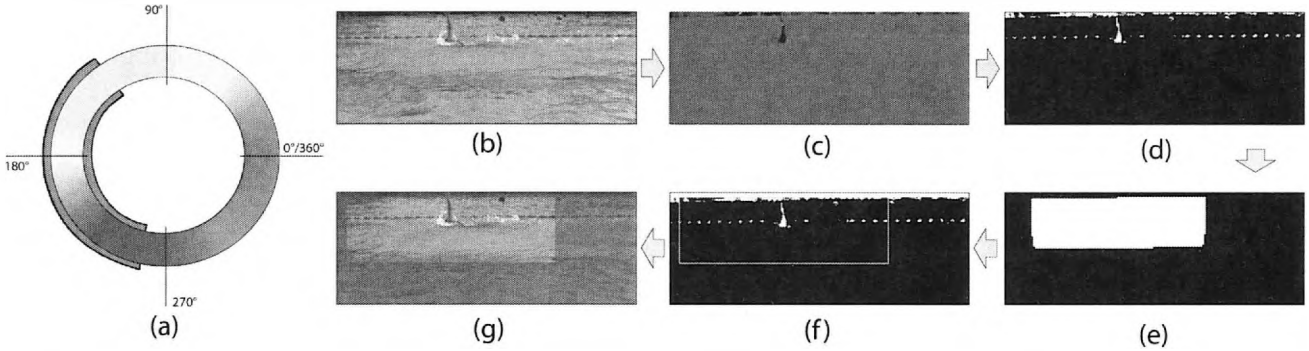

Figure 3. (a) The HSV color wheel with non-skin hue values marked by a red border. (b) Input video frame. (c) The hue values of (b). (d) All skin-colored pixels are represented by white pixels. (e) Areas where the decision rule $t_r(w)$ holds. (f) Slightly extended bounding box around merged areas from (e). (g) Final search area.

template images is computed. We scan each video frame for areas which do not feature a similar ratio in order to exclude such areas in the upcoming matching process. (2) The most likely locations in time and space of swimmers in a given distinctive pose of a given swimming stroke is determined by matching sequences of feature descriptors. This step is the main focus of our research work. We first form template descriptor sequences from mean values of descriptors computed across the template images. These sequences are then matched against sequences from the candidate locations where the features were computed at corresponding relative positions (see Figure 1(c)). We compare different types of feature descriptors. (3) for each pose found the positions of the swimmer's main joints are assigned according to the annotated swimmer's joints in the template images. We then estimate the swimmer's 3D pose by incorporating some prior knowledge about the human body as well as constraints regarding human body proportions and swimming styles. The output are the relative three-dimensional coordinates of the swimmer's main joints (see figure 1(e)).

4. PREFILTERING BY HUE VALUE

We assume that swimmers can show up anywhere in a video sequence. Thus our first step is to fast prune the search to only the most promising areas by removing all areas in each frame which are very unlikely to contain a swimmer.

The HSV color space makes it easy to define a color interval that represents possible human skin colors in and above the water as it represents the color tone by a single hue value. Commonly the hue values are represented by angles ranging from 0° to 360°. Thus we established a color interval $[l_{hue}, u_{hue}] = [90°, 240°]$ for non-skin colors under most lighting conditions and camera configurations. This established color interval range is visualized in Figure 3(a) by the red boundary around the color tones of the hue color ring.

Based on this hue color range, we compute a mask image that marks all possible skin-color pixels by white and all non-skin color pixels by black (see Figure 3(d)). Often image regions free of swimmers will show up mostly black with possibly sporadic white pixels, while regions with swimmers will have a denser set of white pixels. This observation can be exploited to identify irrelevant locations within a video frame.

Skin color ratio: For each training template of size $w = (0, 0, width, height)$ we compute the ratio $r_a(w)$ of the number of skin color pixels to the number of non-skin color pixels, and model areas of swimmers by the average μ_r and the standard deviation σ_r of the observed ratios r_a for the training templates. In other words areas of swimmers are modeled by a Gaussian distribution of the ratio r_a for a test window $w = (x, y, width, height)$. The confidence level is set manually by the constant scaling factor c.Thus, the simple decision rule $t_r(w)$ for a test window w becomes:

$$t_r(w) = \begin{cases} 1 & \text{if } (\mu_r - c\sigma_r) < r_a(w) < (\mu_r + c\sigma_r) \\ 0 & \text{else} \end{cases} \tag{1}$$

Integral images: The search for relevant image regions can be sped-up by employing integral images as introduced by Viola et al.[14] An integral image is an image which stores at position (x, y) the sum of the values of all pixels within the rectangle from the original image's origin $(0, 0)$ to image location (x, y). Transferred to our problem, the integral image I_{int} stores at pixel position (x, y) the number of pixels featuring a hue value from the skin color interval within the rectangle from $(0, 0)$ to (x, y). Thus given the hue image $I_{hue}(x, y)$ of the current video frame, we first compute its skin color mask $I_{hf}(x, y)$:

$$I_{hf}(x, y) = \begin{cases} 0 & \text{if } I_{hue}(x, y) \in [l_{hue}, u_{hue}] \\ 1 & \text{if } otherwise \end{cases} \tag{2}$$

Then the integral image $I_{int}(x, y)$ is computed recursively by

$$I_{int}(x, y) = I_{int}(x, y-1) + I_{int}(x-1, y) + I_{hf}(x, y) - I_{int}(x-1, y-1) \tag{3}$$

with

$$I_{int}(x, y) = 0 \text{ if } x = -1 \text{ or } y = -1. \tag{4}$$

Now the ratio $r_a(w)$ of skin colored to non-skin colored pixels of a region $w = (x, y, width, height)$ can be quickly computed by:

$$r_a(w) = \frac{1}{width \cdot height} SPV_{(x,y,width,height)} \tag{5}$$

where the sum of pixel values $SPV_{(x,y,w,h)}$ is computed using the integral image by:

$$SPV_{(x,y,w,h)} = I_{int}(x-1, y-1) + I_{int}(x+w-1, y+h-1) - I_{int}(x-1, y+h-1) - I_{int}(x+w-1, y-1) \tag{6}$$

Note that this computation only requires four simple lookups from the integral image as opposed to scanning the complete rectangle. Finally all overlapping relevant regions are merged and slightly extended to minimize the accidental removal of relevant areas. Only the relevant regions are further investigated in the subsequent steps to come. Figure 3 illustrates the whole filtering process.

Search space reduction: On our test videos the true positive rate of the hue value prefiltering was about 80%. Hence about 20% of the areas with at least one swimmer were falsely rejected. However, these falsely rejected areas usually showed the swimmer in an unfavorable and unwanted pose (e.g. both arms were under the water) and therefore their exclusion did not pose any pratical problem as key poses were never accidentally discarded. For our sample video frames the search space could be reduced by 50% - 70% on average (see figure 3(g) for an example), depending on the swimmer's size in relation to the pool and on how many skin-colored background objects were present.

5. SWIMMER DETECTION

The remaining search areas are now inspected for finding a swimmer in a predetermined key pose at any scale. Examples of key poses for backstroke, freestyle, and butterfly are shown in figure 2. These nine key pose image also represent our training images for the three desired key poses. They were collected manually.

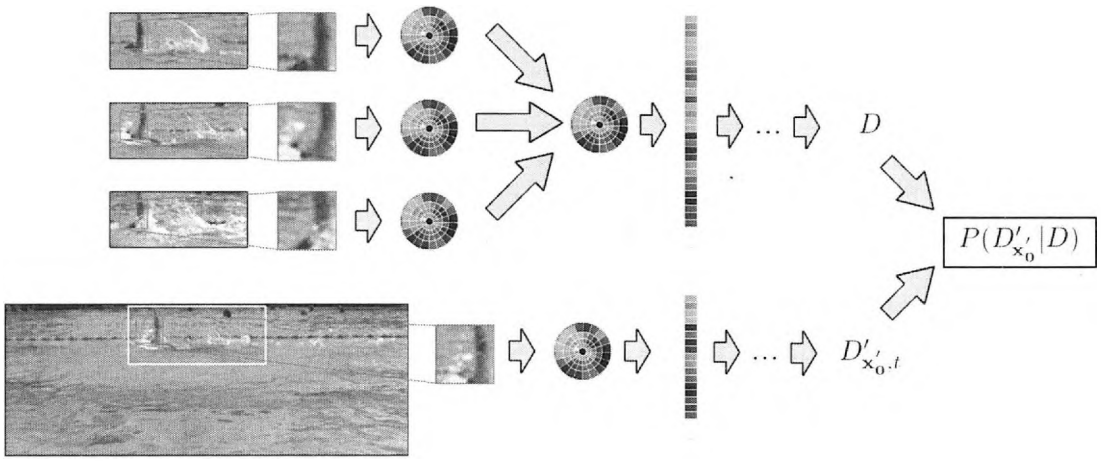

Figure 4. The top portion shows the computation of one mean template descriptor for one location (red square) in three templates. The resulting descriptor is one of the descriptors of the ordered sequence D. The bottom part of the figure depicts one candidate region (yellow rectangle) of one video frame t where the descriptors are computed at corresponding relative locations starting at location $\mathbf{x}'_0$, and form the descriptor sequence $D'_{\mathbf{x}'_0}$. Subsequently, both sequences are matched by computing $P(D'_{\mathbf{x}'_0}|D)$.

5.1 Modeling Key Poses

The first step of our detection process is to create a fix-scale model of a key pose from the template images. Hence we compute feature descriptors on a dense grid inside each template image of the respective swimming stroke. We have tested different types of features: SIFT,[18] Geometric Blur,[19] Self-Similarity,[16] and histograms of oriented gradients (HOG).[20]

For each (relative) grid position $\mathbf{x}$ of a specific stroke we construct a Gaussian descriptor model of the feature vector at that lcoation by computing the mean descriptor vector $\overline{\mathbf{v}}_\mathbf{x}$ over all template images. The top portion of figure 4 illustrates this process. The mean feature vector is considered the mean of a multivariate Gaussian descriptor model $\overline{\mathbf{d}}_\mathbf{x}$ with a constant diagonal covariance matrix $\Sigma_\mathbf{x}$, thus $\overline{\mathbf{d}}_\mathbf{x} = N(\overline{\mathbf{v}}_\mathbf{x}, \Sigma_\mathbf{x})$. We use a constant variance value for the covariance matrix since we do not have enough template images to compute actual covariances.

Computing a mean descriptor $\overline{\mathbf{d}}$ centered at each of the n positions of the grid yields an ordered sequence D of n mean descriptors starting at some position $\mathbf{x}_0$: $D = [\overline{\mathbf{d}}_{\mathbf{x}_0}, ..., \overline{\mathbf{d}}_{\mathbf{x}_{n-1}}]$ where the $\mathbf{x}_\mathbf{i}$ are the grid positions.

5.2 Matching Test Windows with Key Poses

The video is then searched by a sliding window of the same size as the template images of the key poses under search for feature vector sequences which are most similar to the feature vectors of the respective template descriptor sequence. This is done by computing descriptors at the same distance in the test windows as in the template images for the relevant areas of each frame t. The relevant areas were the areas remaining after the filtering process described in section 4.

Thus, for each candidate image region of the same size as the template images, an ordered sequence of feature vectors $D'_{\mathbf{x}'_0,t} = [\mathbf{v}'_{\mathbf{x}'_0,t}, ..., \mathbf{v}'_{\mathbf{x}'_{n-1},t}]$ is obtained with relative descriptor center locations $\mathbf{x}'_0$ to $\mathbf{x}'_{n-1}$ which correspond to the locations of the template descriptors. In other words, position $\mathbf{x}'_\mathbf{i}$ corresponds to template position $\mathbf{x}_\mathbf{i}$. The computation of descriptor sequences from a video frame is illustrated by the lower portion of figure 4.

The likelihood of the region with descriptors $D'_{\mathbf{x}'_0}$ containing the swimmer taking the template pose is modeled by $P(D'_{\mathbf{x}'_0}|D)$. To derive this probability let t_{hit} be a frame where the swimmer takes the template pose and $\mathbf{x}_{hit}$

be the true location of the swimmer inside that frame. Then the detection target $(\mathbf{x}_{hit}, t_{hit})$ can be defined as

$$(\mathbf{x}_{hit}, t_{hit}) = \underset{\mathbf{x}'_0,t}{\mathbf{argmax}}\ P(D'_{\mathbf{x}'_0,t}|D) \tag{7}$$

To obtain the matching probability for the sequence $D'_{\mathbf{x},t}$ and the template sequence D, we use the Gaussian template descriptor model described above which is given by $\overline{\mathbf{d}}_{\mathbf{x}} = N(\overline{\mathbf{v}}_{\mathbf{x}}, \Sigma_{\mathbf{x}})$. Therefore, we compute the matching probability $P(\mathbf{v}'_{\mathbf{x}'_i,t}|\overline{\mathbf{d}}_{\mathbf{x}_i})$ between feature vector $\mathbf{v}'_{\mathbf{x}'_i,t}$ at position $\mathbf{x}'_i$ inside the video frame and the mean template descriptor $\overline{\mathbf{d}}_{\mathbf{x}_i}$ at the corresponding realtive position $\mathbf{x}_i$ as follows:

$$P(\mathbf{v}'_{\mathbf{x}'_i,t}|\overline{\mathbf{d}}_{\mathbf{x}_i}) = \det(2\pi\Sigma_{\mathbf{x}_i})^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{v}'_{\mathbf{x}'_i,t} - \overline{\mathbf{v}}_{\mathbf{x}_i})^T {\Sigma_{\mathbf{x}_i}}^{-1}(\mathbf{v}'_{\mathbf{x}'_i,t} - \overline{\mathbf{v}}_{\mathbf{x}_i})\right) \tag{8}$$

As already mentioned, due to the low number of templates, a diagonal covariance matrix with constant variances σ^2 is used. The probability can thus be written as

$$P(\mathbf{v}'_{\mathbf{x}'_i,t}|\overline{\mathbf{d}}_{\mathbf{x}_i}) = \eta \exp\left(-\frac{1}{2}\frac{\|\mathbf{v}'_{\mathbf{x}'_i,t} - \overline{\mathbf{v}}_{\mathbf{x}_i}\|^2}{\sigma^2}\right) \tag{9}$$

where η is the normalization constant of equation 8.

Having defined the matching probability between two descriptors, we can compute the matching probability between the descriptor sequence from the video frame and the template descriptor sequence by simply assuming independence between descriptors at different locations. Thus the matching probabilities of all corresponding descriptors becomes:

$$P(D'_{\mathbf{x}'_0,t}|D) = \left(\prod_{i=0}^{n-1} P(\mathbf{v}'_{\mathbf{x}'_i,t}|\overline{\mathbf{d}}_{\mathbf{x}_i})\right) \tag{10}$$

As we are dealing with probability values, the product in equation 10 yields extremely small numbers. Fortunately, the logarithm of a term is maximized by the same arguments as the term itself and the logarithm of a product can be written as a sum of logarithms:

$$\underset{\mathbf{x}'_0,t}{\mathbf{argmax}}\ \left(\prod_{i=0}^{n-1} P(\mathbf{v}'_{\mathbf{x}'_i,t}|\overline{\mathbf{d}}_{\mathbf{x}_i})\right) = \underset{\mathbf{x}'_0,t}{\mathbf{argmax}}\ \left(\sum_{i=0}^{n-1} \log\left(P(\mathbf{v}'_{\mathbf{x}'_i,t}|\overline{\mathbf{d}}_{\mathbf{x}_i})\right)\right) \tag{11}$$

The latter term of equation 11 is commonly referred to as the *log-likelihood*. So far we can only detect swimmers matching the size of the swimmers in the template images. However, swimmers may vary in scale due to their distances from the camera and/or overall body sizes. To tackle this problem, we perform *multi-scale search* which means that we scan each frame at different scales. Thus the final goal is to find $(\mathbf{x}_{hit}, t_{hit}, scale_{hit})$.

6. POSE ASSIGNMENT

After determining the swimmer's most likely location in the video frame where she/he takes the template pose, the relative positions of her/his main joints are assigned. The main joints are neck, shoulders, elbows, wrists, hip, thigh joints, knees and ankles. We also estimate the 3D coordinates of these joints.

Average Relative Joint Positions: The relative joint positions are deduced from annotated video frames. All of the main joints' positions were localized manually beforehand in more than 20 overwater and underwater video frames featuring the key pose.[24] The respective joint positions were averaged. As an example figure 5 shows the average relative position of each joint during the key pose of backstroke. These average positions are mapped (i.e., translated and scaled) to the detected position, which is illustrated by figures 1 (c) and (d).

Estimating 3D joint coordinates: Based on the 2D joint positions, we try to estimate 3D coordinates for each joint. For this purpose, we assume scaled orthographic projection which is plausible for the side-view swimming videos we used. This assumption allows us to apply the method introduced by Taylor.[21]

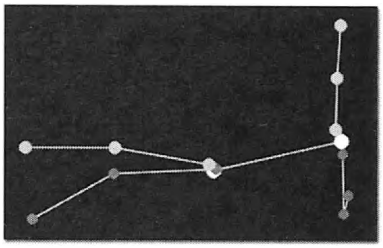

Figure 5. Average joint positions computed from manually annotated video frames showing the backstroke key pose. Yellow dots indicate joints of the swimmer's left side which was in front from the camera's point of view, while red dots are the joints of her/his opposite side. The white dots represent the average positions of the neck and hip respectively.

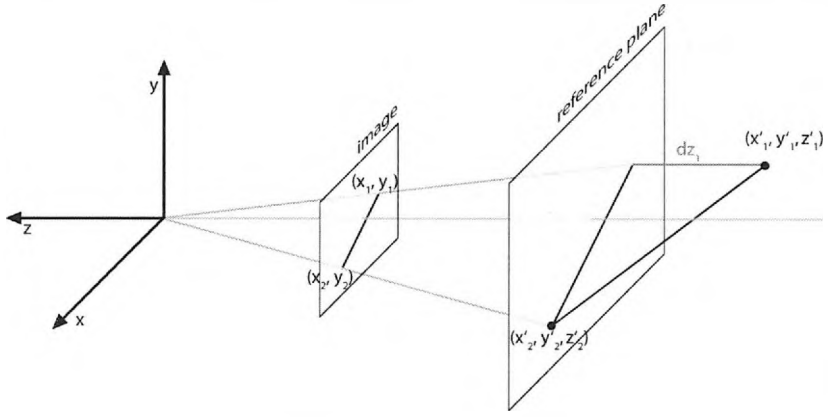

Figure 6. Illustration of scaled orthographic projection: The coordinates of joints in the image are assumed to be on corresponding scaled positions on the reference plane. The z-coordinate is estimated by computing dz.

If we presume scaled orthographic projection, which is illustrated in figure 6, the missing z-coordinate of a joint with 2D coordinates (x_1, y_1) connected to another joint at (x_2, y_2) can be computed as follows:

$$z_1 = z_2' \pm dz_1 \tag{12}$$

with

$$dz_1 = \sqrt{(l'_{1,2})^2 - \frac{(x_1 - x_2)^2 + (y_1 - y_2)^2}{s^2}} \tag{13}$$

where s is the relative scaling factor and $l'_{1,2}$ is the length of the respective limb, which is computed from previously known human limb relations. The z-coordinate of each joint also depends on the estimated z-coordinate of the joint it is connected to, which is denoted by z_2' in equation 12. Each of these kinematic chains start at the positions of the neck or hip, respectively, which we both set to $z = 0$ since we are using side-view videos and the swimmer's spine is supposed to be in-plane.

Note that equation 12 is ambiguous regarding the direction of dz_1 on the z axis of the coordinate system. This ambiguity is solved by introducing constraints which are inferred from previous knowledge about the swimming style and obvious human body proportions. Figure 7 shows estimated three-dimensional joint positions based on the average relative joint positions shown in figure 5, which were assigned to a detection result.

7. EXPERIMENTAL RESULTS

7.1 Feature Types

We compare four different types of features: Self-Similarity, SIFT, Geometric Blur, and HOG.

The *Self-Similarity* descriptor was introduced by Shechtman et al.[16] It models the similarity of a small center image patch to the image patches in its surrounding. These similarities are represented as correlation surfaces

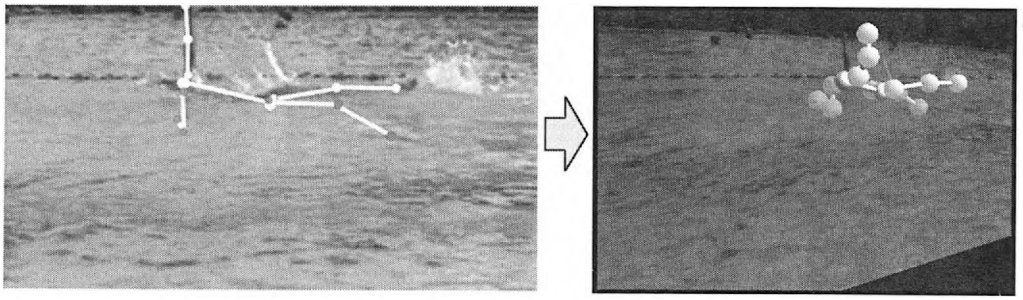

Figure 7. The result of estimating three-dimensional coordinates from the assigned average joint positions in the detection result shown on the left.

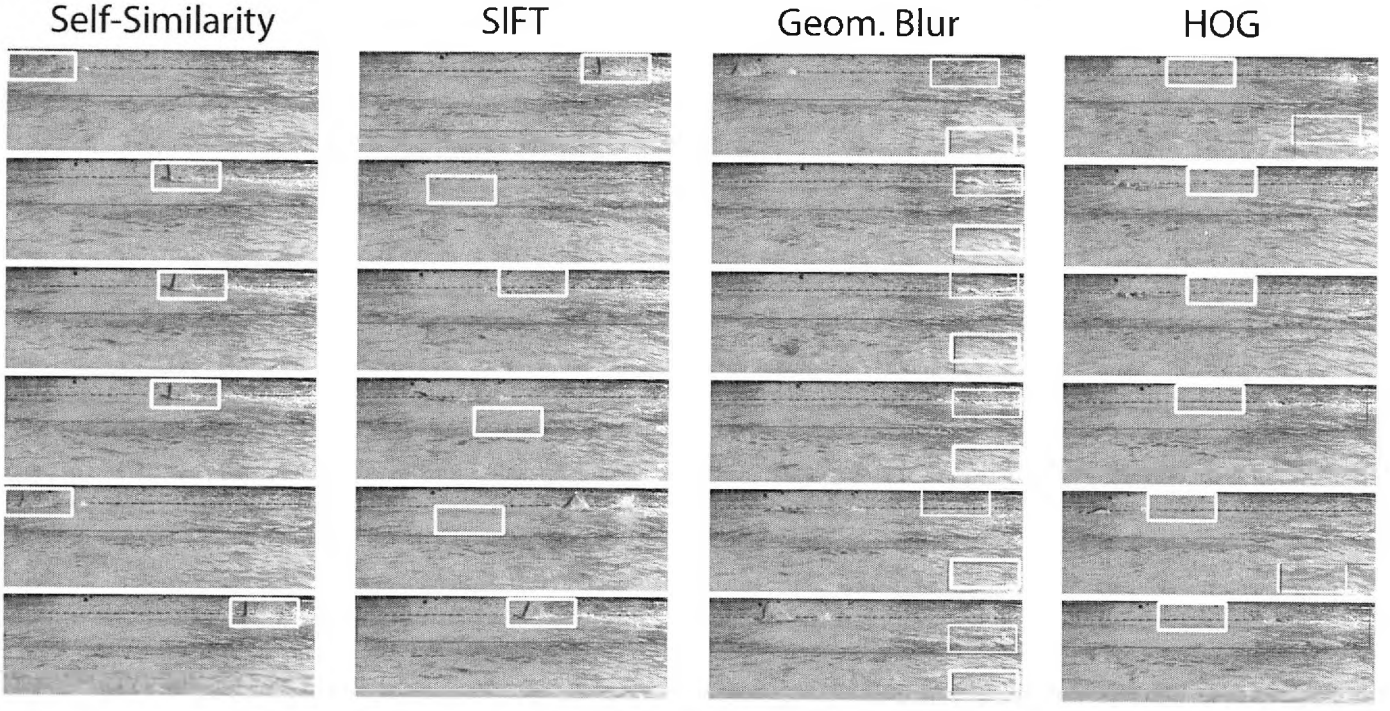

Figure 8. The top six detections for our four features, ordered by descreasing matching probability from top to bottom. The white rectangles represent the best hits in the respective relevant region bounded by a red rectangle. If there are multiple relevant regions inside one frame, the bold rectangle indicates the best hit.

which are then transformed to a log-polar representation (hence the circular appearance of the self-similarity descriptors in the figures of this paper).

The *SIFT* features which were developed by Lowe[18] are among the best-known descriptors. Originally they were invented for localizing key points in order to recognize objects. However we compute them on a dense grid. SIFT features are basically weighted gradient histograms.

The *Geometric Blur* feature was introduced by Berg et al.[19] It models the shapes inside an image region while introducing some invariance towards geometric deformations. This is achieved by blurring the respective image region in proportion to the distance from the region's center.

The last feature we tested were *Histograms of Oriented Gradients* (HOG). Dalal[20] used them to detect people in images and videos.

Figure 8 shows the top six detections in decreasing order of matching probability of each feature type for a video showing a swimmer performing backstroke. The test video consists of a total of 350 frames featuring the key pose and the key pose where the opposite arm is raised four times each. The red rectangles indicate the search areas after filtering by hue value as described in section 4. The template images which were used are shown in figure 2 on the left.

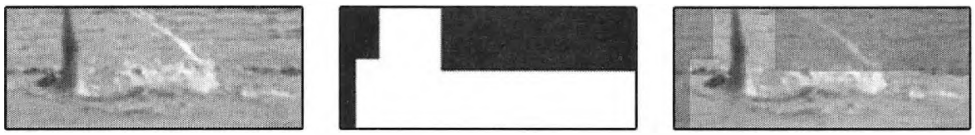

Figure 9. The mask used to determine the regions where descriptors are computed (middle image) .

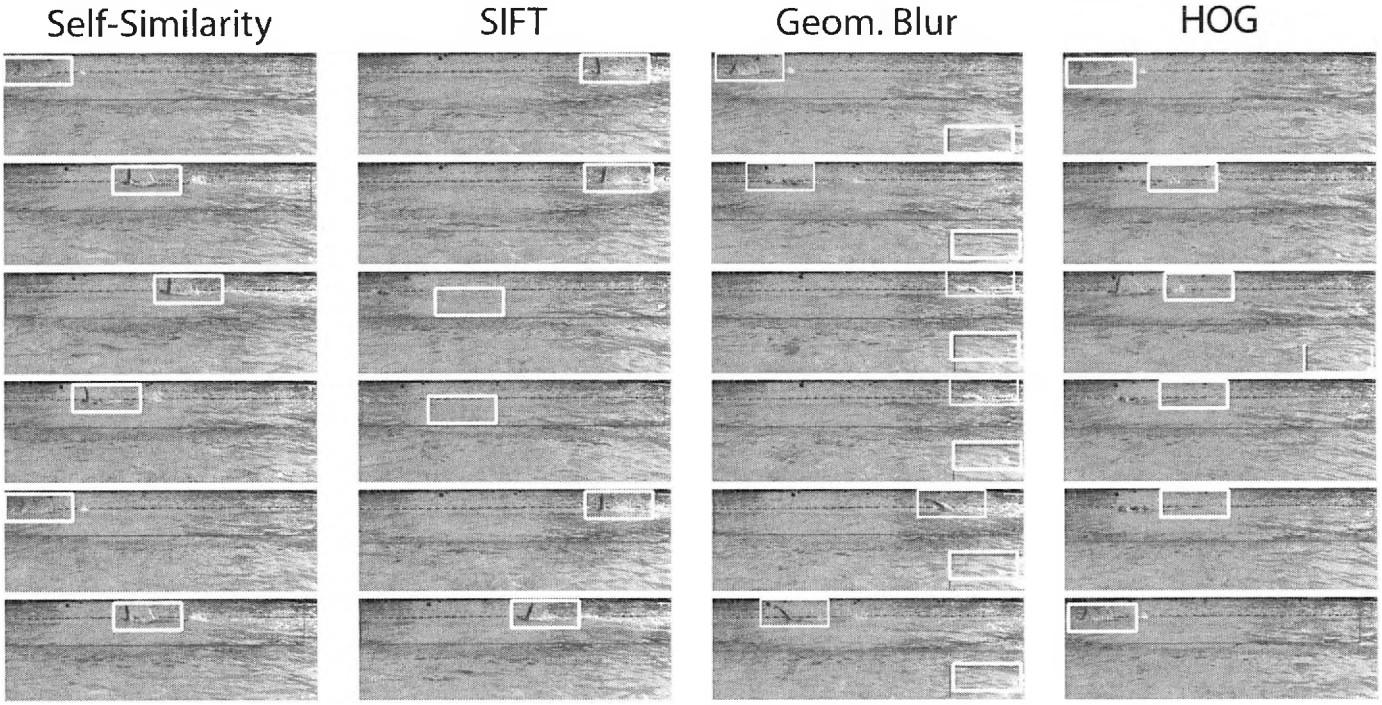

Figure 10. The top six detections for our four features after applying the mask shown in figure 9. Note that the bold rectangles are the best hits for each frame again. Note, the best hits using geometric blur are not the ones containing the swimmer.

According to these results the self-similarity features work best for the test video, while the geometric blur and HOG descriptors appear not to be suitable for this task. However some of the upcoming modifications improve their performances, especially the one of the HOG descriptors. SIFT performs worse than self-similarity. However the top detection can be considered correct. Note that one major issue of the backstroke is that our approach detects the key pose indifferently to the swimmer's visible side, i.e., the backstroke key pose where the swimmer raises her/his left arm is not distinguished from the pose where she/he raises her/his right arm.

7.2 Binary Mask

As our templates include large background areas or areas featuring unpredictable and mostly random splashes (e.g., in the upper right corner of the backstroke templates), we limited the computation of the descriptors to the more relevant and characteristic regions. Therefore we created a binary mask as shown in figure 9. This mask determines by white areas where the descriptors are to be computed. Note that using a mask is not equivalent to cutting out the template images more accurately.

Figure 10 shows the results of using the mask. All descriptors seem to benefit from reducing the templates to the relevant areas. For example, the top hit of HOG can be considered correct now. The reason is most likely that the somewhat random features inside the areas are excluded and thus cannot mask the characteristic features of the swimmer's arm and body position anymore.

We also tried a mask covering the swimmer's arm, head, and about 40% of her/his body as well as a mask computed from HOG feature values, where all locations were excluded if the descriptors value of the strongest component did not surpass a certain threshold. The results for both alternative masks were somewhat worse than the ones shown in figure 10, however still better than the results without any mask.

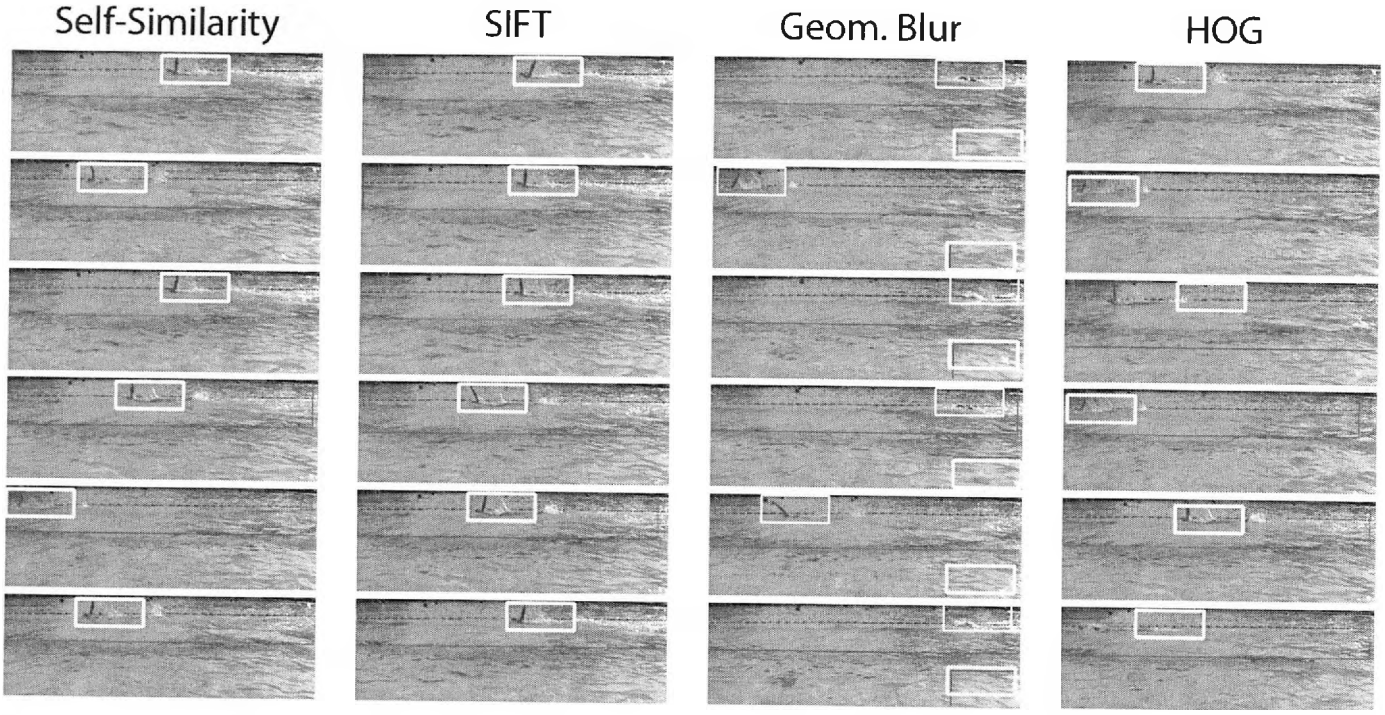

Figure 11. The top six detections for our four feature types using separate template descriptors and the mask shown in figure 9.

7.3 Separate Feature Descriptors

Even our few template training images (three for each key stroke pose throughout our experiments) exhibited some significant visual diversity, which might be better modelled by a Gaussian mixture model. Thus, we alternatively computed separate Gaussian descriptor models for each indidividual template image. Let m be the number of available templates, then at each position $\mathbf{x}$ we obtain m different descriptors now representing a set of m Gaussian models:

$$\mathbf{d_x} = [N\left(\mathbf{v}_\mathbf{x}^1, \Sigma_\mathbf{x}\right), ..., N\left(\mathbf{v}_\mathbf{x}^m, \Sigma_\mathbf{x}\right)] \tag{14}$$

where $\mathbf{v}_\mathbf{x}^j$ with $j \in [1, ..., m]$ is the feature vector computed at position $\mathbf{x}$ in template j. Again we use some constant diagonal covariance matrix $\Sigma_\mathbf{x}$ with a smaller variance value though. Thus instead of one "wide" Gaussian model centered at the mean feature vector at each grid position, we use m "narrow" Gaussian models – one for each template – at each grid position. After computing the descriptors for each grid position within each template image, we once again obtain an ordered sequence of descriptors $D = [\mathbf{d_{x_0}}, ..., \mathbf{d_{x_{n-1}}}]$.

During the matching process, we simply compute a matching probability between each feature vector $\mathbf{v}'_{\mathbf{x}'_\mathbf{i},t}$ and each of the m template descriptors at the corresponding position $\mathbf{x_i}$. We then accept the maximum of the resulting m probabilities as the matching result for position $\mathbf{x_i}$, so we adjust equation 9 as follows:

$$P(\mathbf{v}'_{\mathbf{x}'_\mathbf{i},t}|\mathbf{d_{x_i}}) = \max\left(\eta_1 \exp\left(-\frac{1}{2}\frac{\|\mathbf{v}'_{\mathbf{x}'_\mathbf{i},t} - \mathbf{v}^1_{\mathbf{x_i}}\|^2}{\sigma^2}\right), \ldots, \eta_m \exp\left(-\frac{1}{2}\frac{\|\mathbf{v}'_{\mathbf{x}'_\mathbf{i},t} - \mathbf{v}^m_{\mathbf{x_i}}\|^2}{\sigma^2}\right)\right) \tag{15}$$

The remainder of the matching process is the same as for mean template descriptors which was described in section 5.2.

Figure 11 shows the results for using separate descriptors and the mask introduced in the previous section. The overall result looks somewhat better, since there are more correct or near-correct hits among the top six hits for SIFT and HOG descriptors.

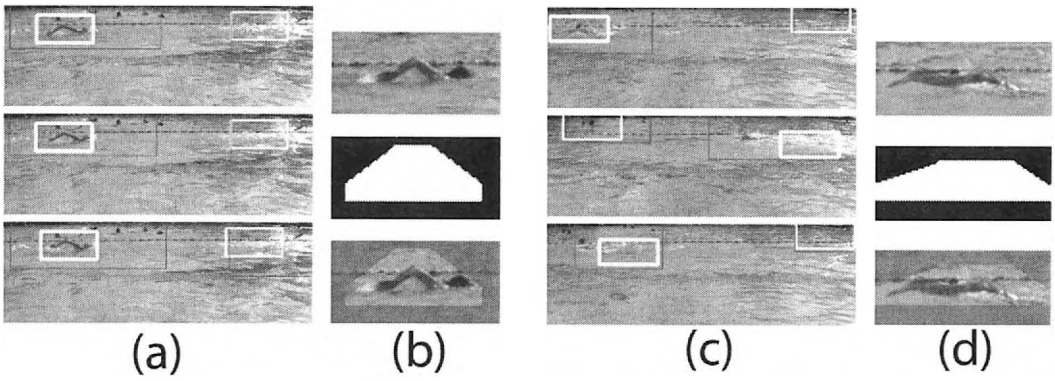

Figure 12. The top three detections for the swimming styles freestyle (a) and butterfly (c) and the respective masks (b) and (d).

7.4 Other Swimming Styles

We also applied our framework to videos featuring swimmers performing freestyle and butterfly using the respective template images shown in figure 2. As a result of the previous experiments, we used self-similarity features, separate template descriptors and binary masks. Again, we searched two test videos of 350 frames each. Figure 12 shows the top three detections for both of these test videos and the respective masks we used. The top detection is correct for both strokes. However, the overall results of our experiments suggest a lack of robustness which is subject to further research.

8. CONCLUSION

In this paper we presented a three step framework for automatic pose initialization of swimmers in videos. We first reduce the search space inside of the target video frames by excluding areas which do not feature a certain ratio of skin to non-skin colored pixels. Then we compute feature descriptors on a dense grid within the video frames and match them to descriptors computed from template images beforehand in order to find the frame regions which are most similar to these templates. The template images show a distinct key pose of the respective swimming style. Finally, we assign three-dimensional joint positions to the detection result based on the two-dimensional average relative joint positions which were obtained from annotated video frames featuring the key pose in advance.

Our experimental results suggest that self-similarity descriptors work best among the feature types SIFT, Self-Similarity, Geometric Blur and HOG. Furthermore, some modifications such as using additionally a binary mask and separate descriptors seem promising. The main reason for employing these extensions is to get the most out of the low number of available template images.

REFERENCES

1. S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape context," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**, pp. 509 – 522, April 2002.
2. P. F. Felsenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *IJCV* **61**, pp. 55–79, October 2005.
3. R. Lienhart, A. Kuranov, and V. Pisarevsky, "Empirical analysis of detection cascades of boosted classifiers for rapid object detection," *MRL Technical Report* **Intel Labs**, May 2002. Revised Dec. 2002.
4. Z. Chen and H.-J. Lee, "Knowledge-guided visual perception of 3-d human gait from a single image sequence," *IEEE Transactions on Systems, Man and Cybernetics* **22**, pp. 336–342, 1992.
5. A. Bharatkumar, K. Daigle, M. Pandy, Q. Cai, and J. K. Aggarwal, "Lower limb kinematics of human walking with the medial axis transformation," *Workshop on Motion of Non-Rigid and Articulated Objects* , pp. 70–76, 1994.

6. M. Leung and Y. H. Yang, "A model based approach to labelling human body outlines," *Proceedings of the 1994 IEEE Workshop on Motion of Non-Rigid and Articulated Objects* , pp. 57–62, 1994.
7. X. Ren, A. C. Berg, and J. Malik, "Recovering human body configurations using pairwise constraints between parts," *ICCV* **1**, pp. 824–831, 2005. Beijing.
8. G. Mori and J. Malik, "Recovering 3d human body configurations using shape contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28**, pp. 1052–1062, July 2006.
9. A. Agarwal and B. Triggs, "3d human pose from silhouettes by relevance vector regression," *IEEE International Conference on Computer Vision and Pattern Recognition* , 2004.
10. M. Andriluka, S. Roth, and B. Schiele, "Pictorial structures revisited: People detection and articulated pose estimation," *IEEE Conference on Computer Vision and Pattern Recognition* , 2009.
11. M. Dimitrijevic, V. Lepetit, and P. Fua, "Human body pose recognition using spatio-temporal templates," *IEEE ICCV Workshop on Modeling People and Human Interaction* , 2005.
12. G.Rogez, J. Rihan, S. Ramalingam, C. Orrite, and P. Torr, "Randomized trees for human pose detection," *IEEE Conference on Computer Vision and Pattern Recognition* , pp. 1–8, 2008.
13. X. Tong, L. Duan, C. Xu, Q. Tan, and H. Lu, "Local motion analysis and its application in video based swimming style recognition," *ICPR* , 2006.
14. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *IEEE CVPR* , pp. 511–518, 2001. Kauai Marriott, Hawaii.
15. R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," *IEEE ICIP* , pp. 900–903, 2002. Rochester, New York.
16. E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," *IEEE Conference on Computer Vision and Pattern Recognition* , 2007.
17. B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19**, pp. 696–710, July 1997.
18. D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision* **60**(2), pp. 91–110, 2004.
19. A. C. Berg and J. Malik, "Geometric blur for template matching," *CVPR* **1**, pp. 607–614, 2001. PhD thesis.
20. N. Dalal, "Finding people in images and videos," *Grenoble* , July 2006.
21. C. J. Taylor, "Reconstruction of articulated objects from point correspondences in a single uncalibrated image," *Computer Vision and Image Understanding* **80**, pp. 677–684, 2000.
22. X. Tong, L. Duan, C. Xu, Q. Tan, H. Lu, J. Wang, and J. S. Jin, "Periodicity detection of local motion," *ICME* , pp. 650–653, 2005. Amsterdam.
23. W.-H. Liao and M.-J. Liu, "Robust swimming style classification from color video," *ICIP* , 2004.
24. T. Greif and R. Lienhart, "An annotated data set for pose estimation of swimmers," Tech. Rep. 2009-18, Institute of Computer Science, University of Augsburg, December 2009.