From Distributed Vision Networks to Human Behavior Interpretation

Hamid Aghajan and Chen Wu

Department of Electrical Engineering
Stanford University, Stanford CA, 94305, USA

Abstract. Analysing human behavior is a key step in smart home applications. Many reasoning approaches utilize information of location and posture of the occupant in qualitative assessment of the user's status and events. In this paper, we propose a vision-based framework to provide quantitative information of the user's posture which can be used to deduct qualitative representations for high-level reasoning. Furthermore, our approach is motivated by potentials introduced by interactions between the vision module and the high-level reasoning module. While quantitative knowledge from the vision network can either complement or provide specific qualitative distinctions for AI-based problems, these qualitative representations can offer clues to direct the vision network to adjust its processing operation according to the interpretation state. The paper outlines potentials for such interactions and describes two vision-based fusion mechanisms. The first employs an opportunistic approach to recover the full-parameterized human model by the vision network, while the second employs directed deductions from vision to address a particular smart home application in fall detection.

1 Introduction

The increasing interest in understanding human behaviors and events in a camera context has heightened the need for gesture analysis of image sequences. Gesture recognition problems have been extensively studied in Human Computer Interactions (HCI), where often a set of pre-defined gestures are used for delivering instructions to machines [1, 2]. However, "passive gestures" predominate in behavior descriptions in many applications. Some traditional application examples include surveillance and security applications, while more novel applications arise in emergency detection in clinical environments [3], video conferencing [4, 5], and multimedia and gaming applications. Some approaches to analyzing passive gestures have been investigated in [6, 7].

In a multi-camera network, access to multiple sources of visual data often allows for making more comprehensive interpretations of events and gestures. It also creates a pervasive sensing environment for applications where it is impractical for the users to wear sensors. Having access to interpretations of posture and gesture elements obtained from visual data over time enables higher-level

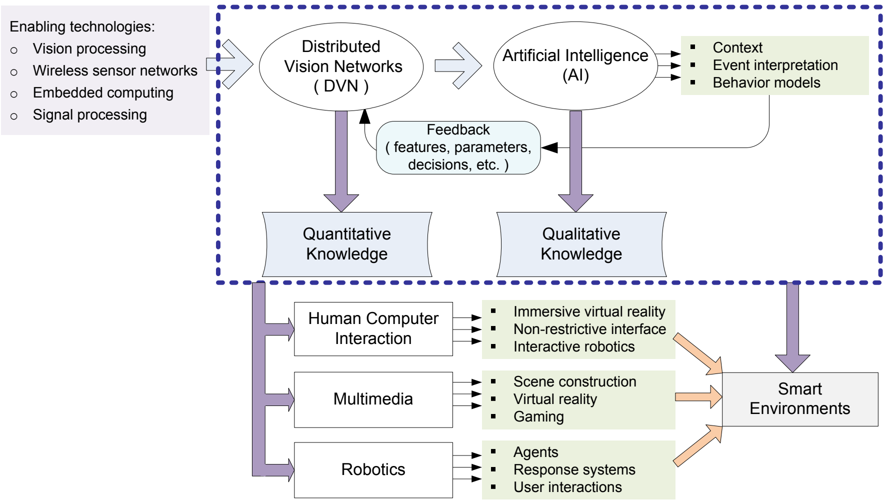

Fig. 1. The relationship between vision networks and high-level AI reasoning, and a variety of novel applications enabled by both.

reasoning modules to deduct the user's actions, context, and behavior models, and decide upon suitable actions or responses to the situation.

Our notion of the role a vision network can play in enabling novel intelligent applications derives from the potential interactions between the various disciplines outlined in Fig. 1. The vision network offers access to quantitative knowledge about the events of interest such as the location and other attributes of a human subject. Such quantitative knowledge can either complement or provide specific qualitative distinctions for AI-based problems. On the other hand, we may not intend to extract all the detailed quantitative knowledge available in visual data since often a coarse qualitative representation may be sufficient in addressing the application [8]. In turn, qualitative representations can offer clues to the features of interest to be derived from the visual data allowing the vision network to adjust its processing operation according to the interpretation state. Hence, the interaction between the vision processing module and the reasoning module can in principle enable both sides to function more effectively. For example, in a human gesture analysis application, the observed elements of gesture extracted by the vision module can assist the AI-based reasoning module in its interpretative tasks, while the deductions made by the high-level reasoning system can provide feedback to the vision system from the available context or behavior model knowledge.

In this paper we introduce a model-based data fusion framework for human posture analysis using opportunistic use of manifold sources of vision-based information obtained from the camera network in a principled way. The framework spans the three dimensions of time (each camera collecting data over time), space (different camera views), and feature levels (selecting and fusing different feature

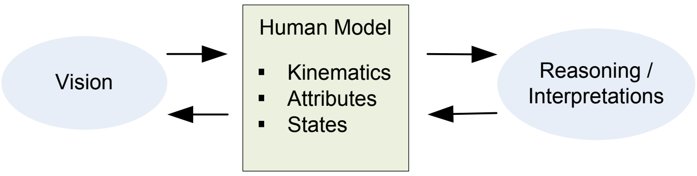

Fig. 2. The human model bridges the vision module and the reasoning module, as the interactive embodiment.

subsets). Furthermore, the paper outlines potentials for interaction between the distributed vision network and the high-level reasoning system.

The structure of the vision-based processing operation has been designed in such a way that the lower-level functions as well as other in-node processing operations will utilize feedback from higher levels of processing. While feedback mechanisms have been studied in active vision areas, our approach aims to incorporate interactions between the vision and the AI operations as the source of active vision feedback. To facilitate such interactions, we introduce a human model as the convergence point and a bridge for the two sides, enabling both to incorporate the results of their deductions into a single merging entity. For the vision network, the human model acts as the embodiment of the fused visual data contributed by the multiple cameras over observation periods. For the AI-based functions, the human model acts as a carrier of all the sensed data from which gesture interpretations can be deducted over time through rule-based methods or mapping to training data sets of interesting gestures. Fig. 2 illustrates this concept in a concise way.

In Section 2 we outline the different interactions between the vision and AI modules as well as the temporal and spatial model-based feedback mechanisms employed in our vision analysis approach. Section 3 presents details and examples for our model-based and opportunistic feature fusion mecahnisms in human posture analysis. In Section 4 an example collaborative vision-based scheme for deriving qualitative assessment for fall detection is described. Section 5 offers some concluding remarks and the topics of current investigation.

2 The Framework

Fig. 3 shows the relationship between the low-level vision processing, which occurs in the camera nodes, the instantaneous state resulting from camera collaboration in the visual domain, and the high-level behavior interpretation which is performed in the AI module. The feedback elements provided by the AI module help the vision processing system to direct its processing effort towards handling the more interesting features and attributes.

The concept of feedback flow from higher-level processing units to the lower-level modules also applies when considering the vision network itself. Within each camera, temporal accumulation of features over a period of time can for

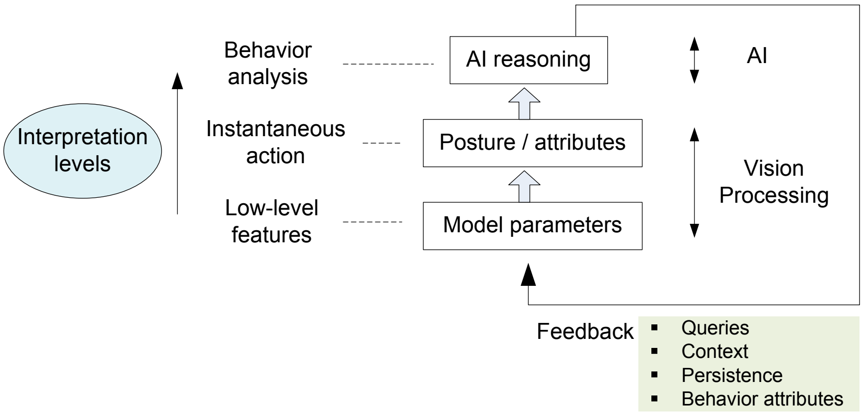

Fig. 3. Interpretation focuses on different levels from vision to behavior reasoning.

example enable the camera to examine the persistence of those features, or to avoid re-initialization of local parameters. In the network of cameras, spatial fusion of data in any of the forms of merged estimates or a collective decision, or in our model-based approach in the form of updates from body part tracking, can provide feedback information to each camera. The feedback can for example be in the form of indicating the features of interest that need to be tracked by the camera, or as initialization parameters for the local segmentation functions. Fig. 4 illustrates the different feedback paths within the vision processing unit.

3 Collaborative Vision Network

We introduce a generic opportunistic fusion approach in multi-camera networks in order to both employ the rich visual information provided by cameras and incorporate learned knowledge of the subject into active vision analysis. The opportunistic fusion is composed of three dimensions, space, time and feature levels. For human gesture analysis in a multi-camera network, spatial collaboration between multi-view cameras naturally facilitates solving occlusions. It is especially advantageous for gesture analysis since human body is self-occlusive. Moreover, temporal and feature fusion help to gain subject-specific knowledge, such as the current gesture and subject appearance. This knowledge is in turn used for a more actively directed vision analysis.

3.1 The 3D Human Body Model

Fitting human models to images or videos has been an interesting topic for which a variety of methods have been developed. Usually assuming a dynamic model (such as walking)[9, 10] will greatly help us to predict and validate the posture estimates. But tracking can easily fail in case of sudden motions or other

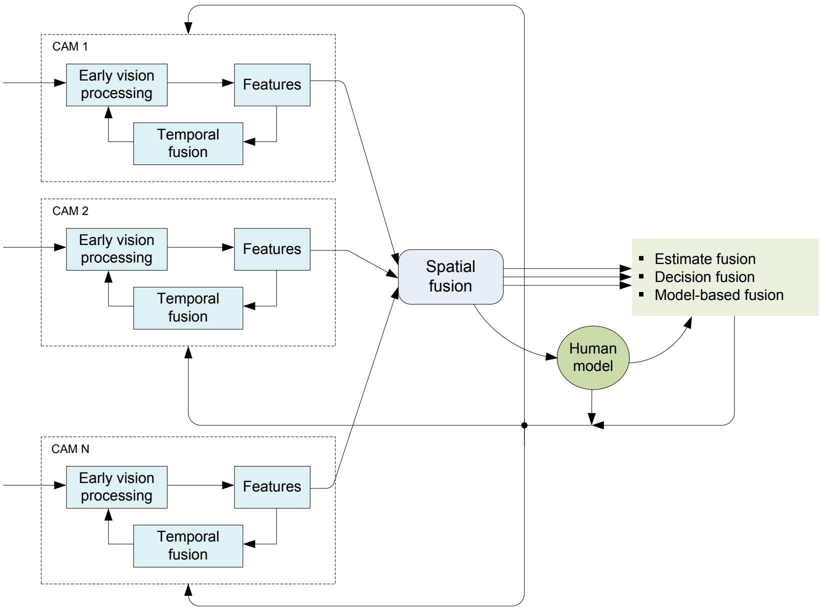

Fig. 4. Different feedback paths within distributed vision processing units.

movements that differ much from the dynamic model. Therefore we always need to be aware of the balance between the limited dynamics and the capability to discover more diversified postures. For multi-view scenarios, a 3D model can be reconstructed by combining observation from different views [11, 12]. Most methods start from silhouettes in different cameras, then points occupied by the subject can be estimated, and finally a 3D model with principle body parts is fit in the 3D space [13]. The approach above is relatively "clean" since the only image component it is based on are the silhouettes. But at the same time the 3D voxel reconstruction is sensitive to the quality of the silhouettes and accuracy of camera calibrations. It is not difficult to find situations where background subtraction for silhouettes suffers for quality or is almost impossible (clustered, complex background, and the subject is wearing clothes with similar colors to the background). Another aspect of the human model fitting problem is the choice of image features. All human model fitting methods are based on some image features as targets to fit the model. Most of them are based on generic features such as silhouettes or edges [14, 12]. Some use skin colors but those methods are prone to failure in some situations since lighting usually has big influence in colors and skin color varies from person to person.

In our work, we aim to incorporate appearance attributes adaptively learned from the network for initialization of segmentation, because usually color or texture regions are easier to find than generic features such as edges. Another emphasis of our work is that images from a single camera are first reduced to short descriptions and then reconstruction of the 3D human model is based on

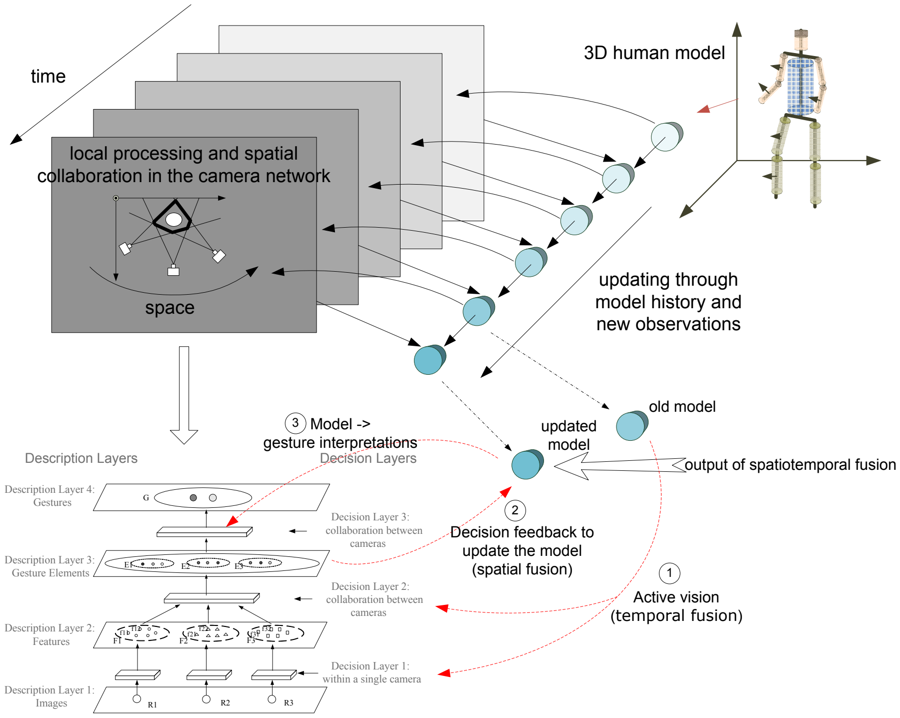
time
local processing and spatial collaboration in the camera network
space
3D human model
updating through model history and new observations
old model
updated model
output of spatiotemporal fusion
3 Model -> gesture interpretations
Description Layers
Decision Layers
Description Layer 4: Gestures
G
Decision Layer 3: collaboration between cameras
Description Layer 3: Gesture Elements
E1
E2
E3
Decision Layer 2: collaboration between cameras
Description Layer 2: Features
f11
f12
f21
f22
f31
f32
F1
F2
F3
Decision Layer 1: within a single camera
Description Layer 1: Images
R1
R2
R3
2 Decision feedback to update the model (spatial fusion)
1 Active vision (temporal fusion)

Fig. 5. Spatiotemporal fusion for human gesture analysis.

descriptions collected from multiple cameras. Therefore concise descriptions are the expected outputs from image segmentation.

In our approach a 3D human body model embodies up-to-date information from both current and historical observations of all cameras in a concise way. It has the following components: 1. Geometric configuration: body part lengths, angles. 2. Color or texture of body parts. 3. Motion of body parts. The three components are all updated from the three dimensions of space, time and features of the opportunistic fusion.

Apart from providing flexibility in gesture interpretations, the 3D human model also plays significant roles in the vision analysis process. First, the total size of parameters to reconstruct the model is very small compared to the raw images, and affordable through communication. For each camera, only segment descriptions are needed for collaboratively reconstructing the 3D model. Second, the model is a converging point of spatiotemporal and feature fusion. All the parameters it maintains are updated from the three dimensions of space, time and features of the opportunistic fusion. In sufficient confidence levels, parameters of the 3D human body model are again used as feedback to aid subsequent vision analysis. Third, although predefined appearance attributes are generally not reliable, adaptively learned appearance attributes can be used to identify the person

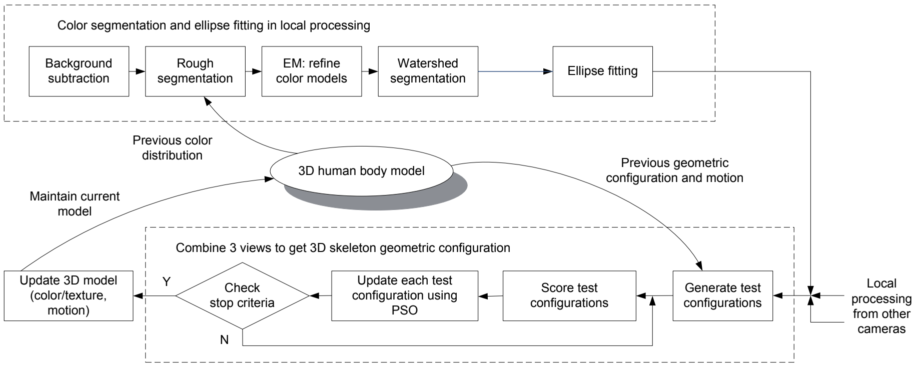
Color segmentation and ellipse fitting in local processing
Background subtraction
Rough segmentation
EM: refine color models
Watershed segmentation
Ellipse fitting
Previous color distribution
3D human body model
Previous geometric configuration and motion
Maintain current model
Combine 3 views to get 3D skeleton geometric configuration
Update 3D model (color/texture, motion)
Y
Check stop criteria
N
Update each test configuration using PSO
Score test configurations
Generate test configurations
Local processing from other cameras

Fig. 6. Algorithm flowchart for 3D human skeleton model reconstruction.

or body parts. Those attributes are usually more distinguishable than generic features such as edges once correctly discovered.

The 3D model maps to the Gesture Elements layer in the layered architecture for gesture analysis (lower left part of Fig. 5) we proposed in [15]. However, here it not only assumes spatial collaboration between cameras, but also connects decisions from history observations with current observations.

3.2 The Opportunistic Fusion Mechanisms

The opportunistic fusion framework for gesture analysis is shown in Fig. 5. On the top of Fig. 5 are spatial fusion modules. In parallel is the progression of the 3D human body model. Suppose now it is t_0, and we have the model with the collection of parameters as M_0. At the next instance t_1, the current model M_0 is input to the spatial fusion module for t_1, and the output decisions are used to update M_0 from which we get the new 3D model M_1.

Now we look into a specific spatial fusion module (the lower part of Fig. 5) for the detailed process. In the bottom layer of the layered gesture analysis, image features are extracted from local processing. Distinct features (e.g. colors) specific for the subject are registered in the current model M_0 and are used for analysis, which may be much easier than always looking for patterns of the generic features (arrow ① in Fig. 5). After local processing, data is shared between cameras to derive for a new estimate of the model. Parameters in M_0 specify a smaller space of possible M_1's. Then decisions from spatial fusion of cameras are used to update M_0 to get the new model M_1 (arrow ② in Fig. 5). Therefore for every update of the model M, it combines space (spatial collaboration between cameras), time (the previous model M_0) and feature levels (choice of image features in local processing from both new observations and subject-specific attributes in M_0). Finally the new model M_1 is used for high-level gesture deductions in a certain scenario (arrow ② in Fig. 5).

An implementation for the 3D human body posture estimation is illustrated in Fig. 6. Local processing in single cameras include segmentation and ellipse fitting for a concise parametrization of segments. For spatial collaboration, ellipses from all cameras are merged to find the geometric configuration of the 3D skeleton model.

3.3 In-Node Feature Extraction

The goal of local processing in a single camera is to reduce raw images/videos to simple descriptions so that they can be efficiently transmitted between cameras. The output of the algorithm will be ellipses fitted from segments and the mean color of the segments. As shown in the upper part of Fig. 6, local processing includes image segmentation for the subject and ellipse fitting to the extracted segments.

We assume the subject is characterized by a distinct color distribution. Foreground area is obtained through background subtraction. Pixels with high or low illumination are also removed since for those pixels chrominance may not be reliable. Then a rough segmentation for the foreground is done either based on K-means on chrominance of the foreground pixels or color distributions from the known model. In the initialization stage when the model hasn't been well established, or when we don't have a high confidence in the model, we need to start from the image itself and use a method such as K-means to find color distribution of the subject. However, when a model with a reliable color distribution is available, we can directly assign pixels to different segments based on the existing color distribution. The color distribution maintained by the model may not be accurate for all cameras, since in different cameras illumination may change. Also the subject's appearance may change due to the movement or lighting conditions. Therefore the color distribution of the model is only used for a rough segmentation in initialization of the segmentation scheme. Then an EM (expectation maximization) algorithm is used to refine the color distribution for the current image. The initial estimated color distribution plays an important role because it can prevent EM from being trapped in local minima.

Suppose the color distribution is a mixture of N Gaussian modes, with parameters $\Theta = \{\theta_1, \theta_2, \ldots, \theta_3\}$, where $\theta_l = \{\mu_l, \Sigma_l\}$ are the mean and covariance matrix of the modes. Mixing weights of different modes are $A = \{\alpha_1, \alpha_2, \ldots, \alpha_3\}$. The EM algorithms aims to find the probability of each pixel x_i belonging to a certain mode θ_l: $Pr(y_i = l|x_i)$.

However, the basic EM algorithm takes each pixel independently, without considering the fact that pixels belonging to the same mode are usually spatially close to each other. In [16] Perceptually Organized EM (POEM) is introduced. In POEM, influence of neighbors is incorporated by a weighting measure $w(x_i, x_j) = e^{-\frac{\|x_i - x_j\|}{\sigma_1^2} - \frac{\|s(x_i) - s(x_j)\|}{\sigma_2^2}}$. $s(x_i)$ is the spatial coordinate of x_i. Then "votes" for x_i from the neighborhood is given by

$$V_l(x_i) = \sum_{x_j} \alpha_l(x_j) w(x_i, x_j), \text{where } {\scriptstyle \alpha_l(x_j) = Pr(y_j = l|x_j)} \tag{1}$$

Then modifications are made to EM steps. In the E step, $\alpha_l^{(k)}$ is changed to $\alpha_l^{(k)}(x_i)$, which means that for every pixel x_i, mixing weights for different modes are different. This is partially due to the influence of neighbors. In the M step, mixing weights are updated by

$$\alpha_l^{(k)}(x_i) = \frac{e^{\eta V_l^{(x_i)}}}{\sum_{k=1}^{N} e^{\eta V_k^{(x_i)}}} \tag{2}$$

η controls the "softness" of neighbors' votes. If η is as small as 0, then mixing weights are always uniform. If η approaches infinity, the mixing weight for the mode with the largest vote will be 1.

After refinement of the color distribution with POEM, we set pixels with high probability (e.g., bigger than 99.9%) that belong to a certain mode as markers for that mode. Then watershed segmentation algorithm is implemented to assign labels for undecided pixels. Finally for every segment an ellipse is fitted to it in order to obtain a concise parameterization for the segment.

3.4 Posture Estimation

Human posture estimation is essentially an optimization problem, in which we try to minimize the distance between the posture and ellipses from multi-view cameras. There can be several different ways to find the 3D skeleton model based on observations from multi-view images. One method is to directly solve for the unknown parameters through geometric calculation. In this method we need to first establish correspondence between points/segments in different cameras, which is itself a hard problem. Common observations for points are rare for human problems, and body parts may take on very different appearance from different views. Therefore it is difficult to resolve ambiguity in 3D space based on 2D observations. A second method would be to cast a standard optimization problem, in which we find optimal θ_i's and ϕ_i's to minimize an objective function (e.g., difference between projections due to a certain 3D model and the actual segments) based on properties of the objective function. However, if the problem is highly nonlinear or non-convex, it'll be very difficult or time consuming to solve. Therefore searching strategies which do not explicitly depend on the objective function formulation are desired.

Motivated by [17], Particle Swarm Optimization (PSO) is used as the optimization technique. The lower part of Fig. 6 shows the estimation process. Ellipses from local processing of single cameras are merged together to reconstruct the skeleton. Here we consider a simplified problem in which only arms change in position while other body parts are kept in the default location. Elevation angles (θ_i) and azimuth angles (ϕ_i) of the left/right upper/lower parts of the arms are specified as parameters. The assumption is that projection matrices from 3D skeleton to 2D image planes are known. This can be achieved either from locations of cameras and the subject, or it can be calculated from some known projective correspondences between the 3D subject and points in the images, without knowing exact locations of cameras or the subject.

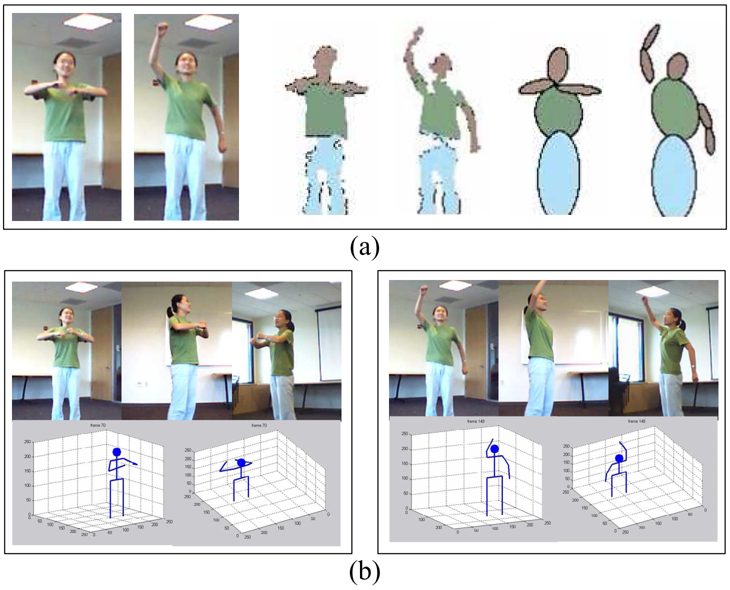

Fig. 7. Examples for gesture analysis in the vision network. (a) In-node segmentation results. (b) Skeleton model reconstruction by collaborative fusion.

PSO is suitable for posture estimation as an evolutionary optimization mechanism. It starts from a group of initial particles. During the evolution of the particles towards an optimal, they are directed to the good position while keep some randomness to explore the search space. Suppose there are N particles (test configurations) x_i, each is a vector of θ_i's and ϕ_i's. v_i is the velocity of x_i. The best position of x_i so far is $\hat{x}_i$, and the global best position of all x_i's so far is g. $f(\cdot)$ is the objective function that we wish to find the optimal position x to minimize $f(x)$. The PSO algorithm is as follows:

1. Initialize x_i and v_i. v_i is usually set to 0, and $\hat{x}_i = x_i$. Evaluate $f(x_i)$ and set $g = \text{argmin} f(x_i)$.
2. While the stop criterion is not satisfied, do for every x_i
   - $v_i \leftarrow \omega v_i + c_1 r_1 (\hat{x}_i - x_i) + c_2 r_2 (g - x_i)$;
   - $x_i \leftarrow x_i + v_i$;
   - If $f(x_i) < f(\hat{x}_i)$, $\hat{x}_i = x_i$; If $f(x_i) < f(g)$, $g = x_i$.

The stop criterion: after updating all N x_i's once, the increase in $f(g)$ falls below a threshold, then the algorithm exits. ω is the "inertial" coefficient, while c_1 and c_2 are the "social" coefficients. r_1 and r_2 are random vectors with each element uniformly distributed on [0,1]. Choice of ω, c_1 and c_2 controls the convergence process of the evolution. If ω is big, the particles have more inertia and tend to keep their own directions to explore the search space. This allows for more chance of finding the "true" global optimal if the group of particles are currently

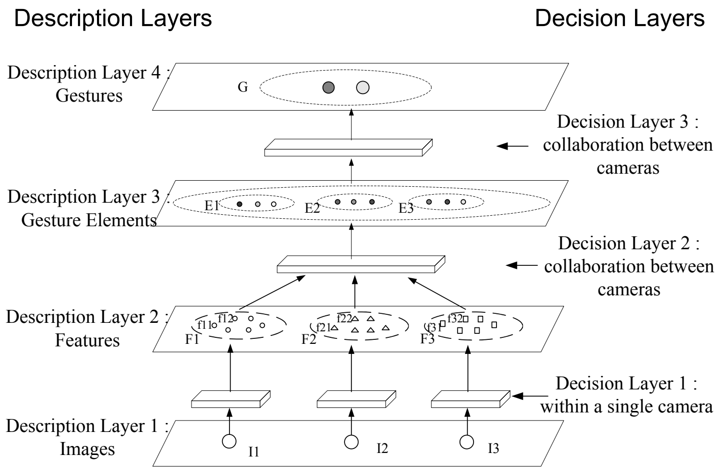

Fig. 8. The layered and collaborative architecture of the gesture analysis system. I_i stands for images taken from camera i; F_i is the feature set for I_i; E_i is the gesture element set in camera i; and G is the set of possible gestures.

around a local optimal. While if c_1 and c_2 are big, the particles are more “social” with the other particles and go quickly to the best positions known by the group. In our experiment, $N = 16$, $\omega = 0.3$ and $c_1 = c_2 = 1$.

Examples for in-node segmentation are shown in Fig. 7(a). Some examples showing images from 3 views and the posture estimates are in Fig. 7(b).

4 Towards Behavior Interpretation

An appropriate classification is essential towards a better understanding of the variety of passive gestures. Therefore, we propose a categorization of the gestures as follows:

- Static gestures, such as standing, sitting, lying;
- Dynamic gestures, such as waving arms, jumping;
- Interactions with other people, such as chatting;
- Interactions with the environment, such as dropping or picking up objects.

Fig. 8 illustrates the layered processing architecture defining collaboration stages between the cameras and the levels of vision-based processing from early vision towards discovery of the gesture elements.

To illustrate the process of achieving high-level reasoning using the collaborative vision-based architecture, we consider an application in assisted living, in which the posture of the user (which could be an elderly or a patient) is monitored during daily activities for detection of abnormal positions such as lying down on the ground. Each of the cameras in the network employs local vision processing on its acquired frames to extract the silhouette of the person. A second level of processing employs temporal smoothing combined with shape fitting

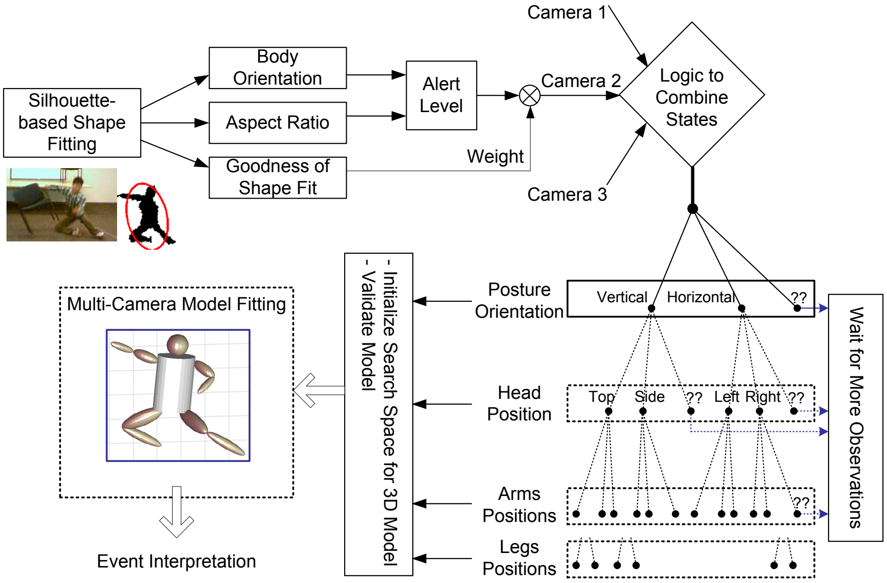

Fig. 9. A tree-based reasoning technique for fall detection. Qualitative descriptions can trace down the branches for specific event detection. The specific deductions can also be feedback for posture reconstruction.

to the silhouette and estimates the orientation and the aspect ratio of the fitted (e.g. elliptical) shape. The network's objective at this stage is to decide on one of the branches in the top level of a tree structure (see Fig. 9) between the possible posture values of *vertical*, *horizontal*, or *undetermined*. To this end, each camera uses the orientation angle and the aspect ratio of the fitted ellipse to produce an *alert level*, which ranges from -1 (for safe) to 1 (for danger). Combining the angle and the aspect ratio is based on the assumption that nearly vertical or nearly horizontal ellipses with aspect ratios away from one provide a better basis for choosing one of the *vertical* and *horizontal* branches in the decision tree than when the aspect ratio is close to one or when the ellipse has for example, a 45-degree orientation.

Fig. 10 illustrates an example of the alert level function combining the orientation and aspect ratio attributes in each camera. The camera broadcasts the value of this function for the collaborative decision making process. Along with the alert level, the camera also produces a figure of merit value for the shape fitted to the human silhouette. The figure of merit is used as a weighting parameter when the alert level values declared by the cameras are combined.

Fig. 11 presents cases in which the user is walking, falling and lying down. The posture detection outcome is superimposed on the silhouette of the person for each camera. The resulting alert levels and their respective weights are shared by the cameras, from which the overall alert level shown in the figure is obtained.

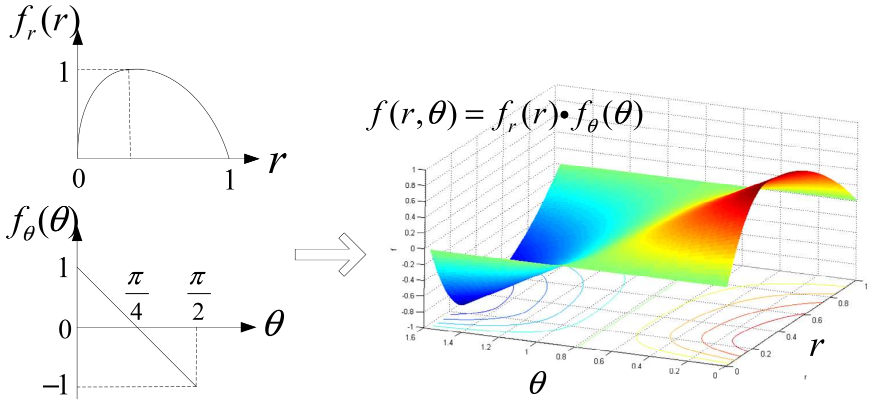

Fig. 10. The alert level functions based on the aspect ratio and the orientation angle of fitted ellipses.

5 Conclusions

In this paper we explore the interactive framework between vision and AI. While vision is helpful to derive reasoning building blocks for higher levels, there is more in the framework. We claim that the feedback between the vision module and the reasoning module is able to benefit both.

A framework of data fusion in distributed vision networks is proposed. Motivated by the concept of opportunistic use of available information across the different processing and interpretation levels, the proposed framework has been designed to incorporate interactions between the vision module and the high-level reasoning module. Such interactions allow the quantitative knowledge from the vision network to provide specific qualitative distinctions for AI-based problems, and in turn, allows the qualitative representations to offer clues to direct the vision network to adjust its processing operation according to the interpretation state. Two vision-based fusion algorithms were presented, one based on reconstructing the full-parameterized human model and the other based on a sequence of direct deductions about the posture elements in a fall detection application.

The current work includes incorporation of body part motion into the full-parameterized human body model allowing the model to carry the gesture elements in interactions between the vision network and the high-level reasoning module. Other extensions of interest include creating a link from the human model to the reduced qualitative description set for a specific application, and utilizing deductions made by the AI system as a basis for active vision in multi-camera settings.

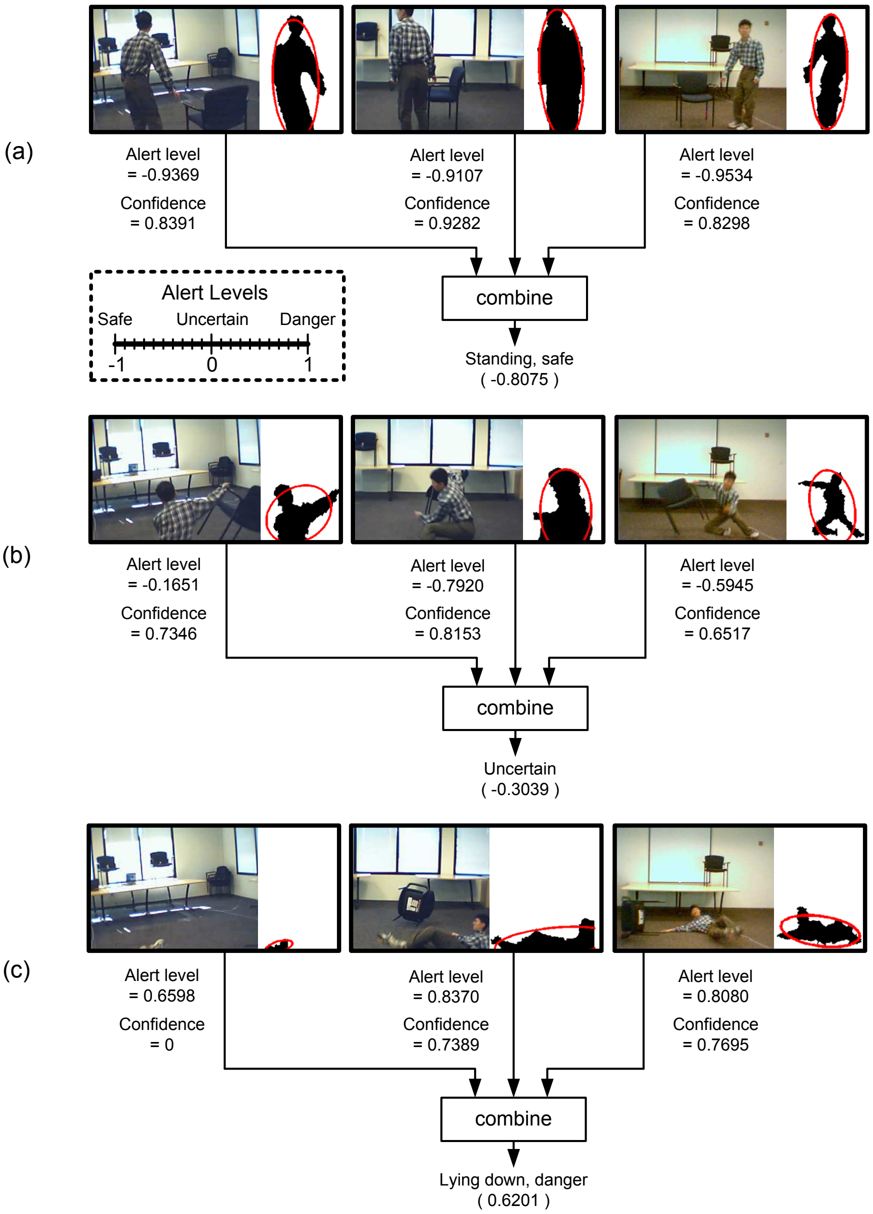

Fig. 11. Three sets of examples from three cameras of different views for fall detection. (a) standing; (b) falling; (c) lying on the ground. Alert levels and their confidence levels are shown. After combining observations from the three cameras a final score is given indicating whether the person is standing (safe) or lying (danger).

References

1. Kwolek, B.: Visual system for tracking and interpreting selected human actions. In: WSCG. (2003)
2. G. Ye, J. J. Corso, and G. D. Hager: 7: Visual Modeling of Dynamic Gestures Using 3D Appearance and Motion Features. In: Real-Time Vision for Human-Computer Interaction. Springer-Verlag (2005) 103–120
3. Aghajan, H., Augusto, J., Wu, C., McCullagh, P., , Walkden, J.: Distributed vision-based accident management for assisted living. In: ICOST 2007, Nara, Japan
4. Patil, R., Rybski, P.E., Kanade, T., Veloso, M.M.: People detection and tracking in high resolution panoramic video mosaic. In: Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS). Volume 1. (Oct. 2004) 1323–1328
5. Robertson, C., Trucco, E.: Human body posture via hierarchical evolutionary optimization. In: BMVC06. (2006) III:999
6. Rittscher, J., Blake, A., Roberts, S.: Towards the automatic analysis of complex human body motions. Image and Vision Computing (12) (2002) 905–916
7. Cucchiara, R., Prati, A., Vezzani, R.: Posture classification in a multi-camera indoor environment. In: ICIP05. (2005) I: 725–728
8. Björn Gottfried, Hans Werner Guesgen, and Sebastian Hübner: Spatiotemporal Reasoning for Smart Homes. In: Designing Smart Homes. Springer (2006) 16–34
9. Sidenbladh, H., Black, M.J., Sigal, L.: Implicit probabilistic models of human motion for synthesis and tracking. In: ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part I, London, UK, Springer-Verlag (2002) 784–800
10. Deutscher, J., Blake, A., Reid, I.: Articulated body motion capture by annealed particle filtering. (2000) II: 126–133
11. Cheung, K.M., Baker, S., Kanade, T.: Shape-from-silhouette across time: Part ii: Applications to human modeling and markerless motion tracking. International Journal of Computer Vision **63**(3) (August 2005) 225 – 245
12. Ménier, C., Boyer, E., Raffin, B.: 3d skeleton-based body pose recovery. In: Proceedings of the 3rd International Symposium on 3D Data Processing, Visualization and Transmission, Chapel Hill (USA). (june 2006)
13. Mikic, I., Trivedi, M., Hunter, E., Cosman, P.: Human body model acquisition and tracking using voxel data. Int. J. Comput. Vision **53**(3) (2003) 199–223
14. Sidenbladh, H., Black, M.: Learning the statistics of people in images and video. **54**(1-3) (August 2003) 183–209
15. Wu, C., Aghajan, H.: Layered and collaborative gesture analysis in multi-camera networks. In: ICASSP. (Apr. 2007)
16. Weiss, Y., Adelson, E.: Perceptually organized em: A framework for motion segmentaiton that combines information about form and motion. Technical Report 315, M.I.T Media Lab (1995)
17. Ivecovic, S., Trucco, E.: Human body pose estimation with pso. In: IEEE Congress on Evolutionary Computation. (2006) 1256–1263