Available online at www.sciencedirect.com
ScienceDirect
Computer Vision and Image Understanding 106 (2007) 205–219
Computer Vision and Image Understanding
www.elsevier.com/locate/cviu

Coalitional tracking

Jonathan Dowdall [a], Ioannis T. Pavlidis [a,*], Panagiotis Tsiamyrtzis [b]

[a] Department of Computer Science, University of Houston, Houston, TX 77204, USA
[b] Department of Statistics, Athens University of Economics and Business, 76 Patission Street, Athens 10434, Greece

Received 7 December 2005; accepted 21 August 2006
Available online 29 December 2006
Communicated by James Davis and Riad Hammoud

Abstract

We propose a novel tracking method that uses a network of independent particle filter trackers whose interactions are modeled using coalitional game theory. Our tracking method is general, it maintains pixel level accuracy, and can negotiate surface deformations and occlusions. We tested our method on a substantial video set featuring non-trivial motion from over 40 objects in both the infrared and visual spectra. The coalitional tracker demonstrated fault tolerant behavior that exceeds by far the performance of single particle filter trackers. Our method represents a shift from the typical tracking paradigms and may find application in demanding imaging problems across the electromagnetic spectrum.

Keywords: Face tracking; Coalitional game theory; Template matching; Thermography

1. Introduction

The extraction of high-level information from video through the use of computer vision algorithms has become increasingly important over the past decade. A diverse array of applications use this technology including quality control in the manufacturing sector [1,2], surveillance in the security industry [3,4], biomedical measurements for healthcare [5–7], and behavioral analysis [8–10]. Of key importance to all these computer vision applications is the ability to detect and track objects in video streams.

The problem of tracking can be cast as guessing how things change over time. Specifically, tracking involves modeling how the parameters of the object modulate in successive input frames by using prior knowledge. When this is done accurately, it can be useful in a number of applications where knowing the current state of a given target object is important.

An intriguing line of computer vision research focuses on measurements of physiological signals on facial tissue. The measurements are performed on infrared imagery and are used in biomedical [5–7] and behavioral applications [8–10]. Although, a large body of work has been devoted to facial tracking research [11–13], we found the existing methods insufficient to achieve the high degree of accuracy required in imaging measurements of facial tissue. This was our initial motivation in exploring a novel tracking paradigm.

1.1. Prior work

Computer vision tracking has been dominated by sequential Monte Carlo methods (particle filtering) [14] for the last several years. Among the most popular particle filter tracking methods is the CONDENSATION algorithm, which was introduced by Isard et al. circa 1998 [15–17].

An interesting tracking methodology based on deformable templates was also developed in parallel. Typical deformable templates focus on tracking object contours not surfaces [18]. Therefore, they cannot adequately address out of plane tracking, like the case of left–right facial rotation.

Alternative tracking methodologies employ specific models of the target to provide better accuracy [19–21].

* Corresponding author.
E-mail address: ipavlidi@central.uh.edu (I.T. Pavlidis).

1077-3142/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2006.08.011

Unfortunately, this increased accuracy comes at the expense of speed and generality. A noteworthy modeling approach is known as active appearance modeling and it takes into account both shape and texture [22,23]. For example, Dornaika et al. [23] first recover the 3D head pose using a deformable wire-frame and then, local motion associated with some facial features using active appearance model search. Such 3D active appearance models can potentially perform quality tracking in demanding facial imaging applications in the visual spectrum. However, their performance may break in thermal infrared imagery due to thermal diffusion and the resulting fuzzy image edges. In such an environment, appearance models may have a hard time maintaining 3D–2D correspondences, which are partly based on thermal gradients.

Tracking in the thermal infrared spectrum is of particular interest to us because recent research demonstrated that many vital signs including blood flow [5], pulse [6], and breathing function [7] can be measured in this modality. The success of these measurements depends strongly on a reliable tracking method to register the motion of facial tissue.

Our method aims to achieve what sophisticated modeling methods reportedly achieve, but it is more general and robust. It does not employ a single explicit 3D model, but many generic and cooperating 2D particle filter trackers, which are spatially distributed over the target's surface. Our effort can be seen as a first step towards developing a tracking methodology that is able to accurately track a wide array of targets across imaging modalities.

There has been some other work on multiple trackers that work together to follow *multiple objects* [24–27]. In contrast, we employ multiple trackers to track a *single object*.

In the remainder of the paper, we first discuss the details of our tracking approach in Section 2. Then, we outline our experimental design in Section 3, followed by the experimental results in Section 4. Finally, we present our conclusions in Section 5.

2. Tracking methodology

Our goal is to develop a *general* tracking methodology that can *accurately* monitor the motion of the target's *surface* even in the presence of deformation or partial occlusion. Many existing general tracking methods *approximately* monitor the target's *outline*, not its surface. This is a different and far easier problem.

We arrived at a fault tolerant surface tracking method that works both on infrared and visual video without resorting to explicit modeling. It uses a network of particle filter trackers that influence each other (see Fig. 1). Each individual tracker is unreliable at times, but the combination of many neighboring trackers produces robust performance. The inter-tracker influence is modeled as a coalitional game where each tracker is a player, and the goal of the game is to propagate one's influence in subsequent frames of video. Within this framework, the winning coalition of trackers is used to calculate the state of the tracked object.

2.1. Tracking network

We use a network of trackers to achieve accurate surface tracking and fault tolerance. Tracking is maintained even if all but one of the trackers fail in the tracking network. The trackers are each assigned a different portion of the target's surface to track (see Fig. 2). By default the trackers are configured in a regular grid, although alternative configurations are possible through a feature selection mechanism. Inter-tracker communication allows trackers that are correctly tracking the target to "tip-off" other trackers that have become lost, as to the true location of the target's surface. This inter-tracker influence is realized within a statistical framework and is managed by the coalitional game model described in Section 2.2.

The idea arose naturally in the effort to address the problem of facial tissue tracking in the infrared. As the subject's head moves (e.g., left and right) part of the facial

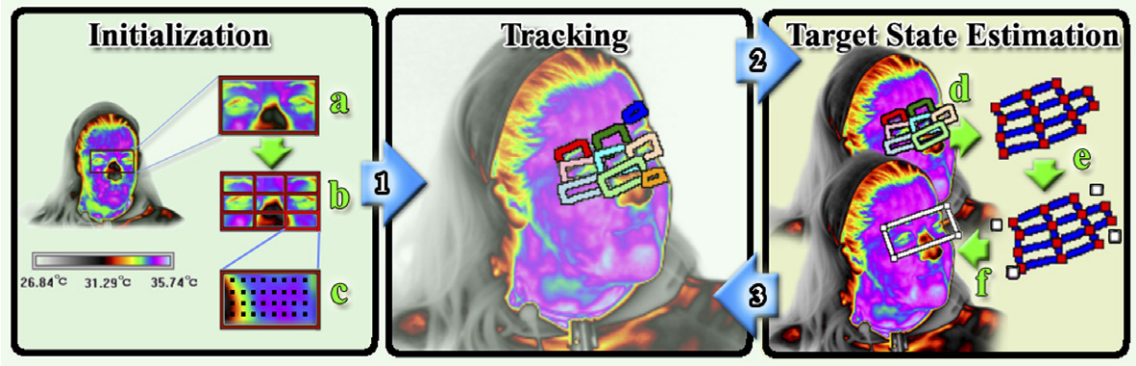

Fig. 1. Overview of the tracking method. *Initialization*: consists of the following steps: (a) extraction of the user selected region of interest from the input video, (b) subdivision of the ROI into the tracking network, and (c) individual tracker template creation. *Tracking*: the individual trackers in the tracking network follow their targets. *Target State Estimation*: consists of the following steps: (d) the winning coalition is produced, (e) the deformation mesh is calculated from the winning coalition, and (f) the deformation mesh is used to calculate the target state. The method proceeds from initialization to tracking (arrow 1) to target state estimation (arrow 2) and back to tracking (arrow 3). In the latter transition, the winning coalition is passed back to the tracking stage to distribute the inter-tracker influence.

Fig. 2. Example of a 3×3 tracking network on a visual image. Each tracker in the network is shown in a different color. Each tracker is tracking a separate part of the target. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

surface is occluded at times. Trackers that correspond to the occluded part of the face are aided by trackers that correspond to the exposed part.

In our implementation, each tracker in the tracking network is an individual particle filter tracker. We denote the state of each individual tracker i at time t by $\mathbf{x}_i^{(t)}$ and its associated image observation by $\mathbf{z}_i^{(t)}$. The target tracker's prior, $p(\mathbf{x}_i^{(t)}|\mathbf{z}_i^{(t-1)})$ will be formed using intra-samples $\mathbf{s}_{(i,i)}^{(t)}$ from tracker i and inter-samples $\mathbf{s}_{(i,j)}^{(t)}$ that correspond to the (inter-tracker) influence of tracker i from tracker j. The inter-samples $\mathbf{s}_{(i,j)}^{(t)}$ are generated based on the initial relationship between the trackers involved in the exchange:

$$\mathbf{s}_{(i,j)}^{(t)} = \mathbf{T}_{\mathbf{x}_i^{(0)}}^{\mathbf{x}_j^{(0)}} \mathbf{x}_j^{(t)}, \tag{1}$$

where $\mathbf{s}_{(i,j)}^{(t)}$ is the inter-sample generated by tracker j for tracker i and $\mathbf{T}_{\mathbf{x}_i^{(0)}}^{\mathbf{x}_j^{(0)}}$ is the transformation that gives a sample for tracker i given a state for target j at time t. The transformation $\mathbf{T}_{\mathbf{x}_i^{(0)}}^{\mathbf{x}_j^{(0)}}$ is computed during initialization for every possible tracker pair $(\mathbf{x}_j^{(0)}, \mathbf{x}_i^{(0)})$.

2.2. Coalitional game

The tracking network is a versatile architecture for tracking objects, but it does not have any intrinsic method to generate the final target state or to manage tracker interaction. The simplest solution would be to allow every tracker to influence all of the other trackers. Unfortunately, this is not an optimal solution, because trackers that have lost their target would be allowed to influence other trackers in the network that have not gone awry. This also highlights the problem of determining which trackers in the network are correctly tracking their targets and which ones have strayed away. What is needed is a mechanism that can determine the validity of each of the trackers, compute the target's state vector based on the valid trackers, and finally propagate the influence of the valid trackers to keep the network correctly tracking the target surface.

There are many optimization algorithms one can use to manage the network of trackers. We chose to optimize tracker interaction using a game theoretic solution for two main reasons: it naturally fits the problem space and it is relatively simple. Game theory [28–31] has been successfully used to analyze topics ranging from simple deterministic games, to complex economic models [32,33], and even international negotiations [34,35]. Our adaptation was to view the trackers as players in a cooperative game [36,37], where the objective was to increase their influence by forming coalitions with other trackers. The winning coalition would then be used to compute the state vector of the target and subsequently propagate its influence onto the entire tracking network.

Specifically, the members $m_j^{(t)}$ of the winning coalition C^t influence every other tracker i in the tracking network by adding inter-samples $\mathbf{s}_{(i,j)}^{(t)}$. Trackers who are not members of the winning coalition cannot propagate any influence at all.

The intuitive affinity of the problem space to cooperative gaming is apparent in the example of facial tissue tracking. There, the winning coalition is composed mostly of trackers that correspond to the exposed part of the face. These are trackers that feature high-quality information and give a 'helping hand' (influence) to the 'clueless' trackers that correspond to the occluded or deformed part of the face.

The coalitional form of an N-tracker game is given by the pair (Ω, Π), where $\Omega = \{1, 2, \ldots, N\}$ is the set of trackers and Π is a real-valued function, called the characteristic function of the game, defined on the set of all coalitions (subsets of Ω), which has cardinality 2^N, and satisfying $\Pi(\varnothing) = 0$ [28]. In other words, the empty set has value zero. The size of a coalition C will be denoted from now on by k, where $k \in \{1, 2, \ldots, N\}$, and there are $\binom{N}{k}$ coalitions of size k. The quantity $\Pi(C_k)$ may be considered as the value, or worth, or power, of coalition $C_k \subset \Omega$ when its members act together as a unit.

The definition of a coalitional game is quite general and leaves the specification of the characteristic function to the game designer. We designed a characteristic function for the tracking game that encompasses four scores. These scores are calculated from the trackers participating in the coalition under consideration at time t:

- Template match $\alpha^{(t)}$.
- Geometric alignment $\beta^{(t)}$.
- Inter-frame projection agreement $\gamma^{(t)}$.
- Inter-frame membership retention $\delta^{(t)}$.

The characteristic scores support the fact that quality tracking is characterized by consistency in the content and geometric configuration of the individual trackers. Specifically, the template match score rewards trackers that

maintain consistent imaging content. The geometric alignment score favors coalitions whose members have geometric alignment analogous to the original ($t = 0$) configuration. The inter-frame projection agreement score is a continuity constraint. It improves robustness by penalizing abrupt (and improbable) changes of the projected state of the target between successive frames. The inter-frame membership retention score is also a continuity constraint. It reflects the tendency of the winning coalition from the previous time step to retain its members.

The template match score $\alpha_{C_k}^{(t)}$ for a coalition C_k of size k at time t is given by:

$$\alpha_{C_k}^{(t)} = \frac{1}{k}\sum_{i=1}^{k} \alpha_{m_i}^{(t)}, \tag{2}$$

where $\alpha_{m_i}^{(t)}$ refers to the template match score (a number in $[0,1]$, see Section 2.6) of member m_i in the coalition C_k at time t.

For the second and third scores we first need to define the function that measures the geometric alignment between two target projections (see Fig. 3), as are computed from samples $\mathbf{s}_i$ and $\mathbf{s}_j$:

$$\begin{aligned} G(\mathscr{F}(\mathbf{s}_i), \mathscr{F}(\mathbf{s}_j)) &= G(\mathbf{S}_i, \mathbf{S}_j) \\ &= \omega \times \left[1 - \frac{((S_{ix} - S_{jx})^2 + (S_{iy} - S_{jy})^2)^{1/2}}{M_d}\right] \\ &\quad + (1-\omega) \times \left[1 - \frac{|S_{i\theta} - S_{j\theta}|}{M_\theta}\right], \end{aligned} \tag{3}$$

where $\mathscr{F}(\mathbf{s})$ is a function that transforms the tracker sample $\mathbf{s}$ into its corresponding target projection $\mathbf{S}$, (S_{ix}, S_{iy}) are the (x, y) coordinates of the center of target projection $\mathbf{S}_i$, while $S_{i\theta}$ is the angle of rotation about the center of target projection $\mathbf{S}_i$. M_d is set to the maximum movement allowed by the target in a single frame while M_θ is the (positive) maximum rotation allowed by the target in a single frame. The weight ω penalizes appropriately the center and angle discrepancies. Ideally, the target projections in Fig. 3(c) should have coincided (perfect alignment), so that the combined projection of the two tracker samples is reminiscent of the original target shape. Note that the upper bound for $G(.,.)$ is 1 (when the two target projections are identical), but the lower bound is not necessarily 0. This would have been the case, if we chose M_d and M_θ to be the maximum observed values at time t, but this would have slowed down the computation. Besides, we do not mind giving negative scores to some tracker pairs (i.e., penalizing as opposed to rewarding them), whose geometric alignment is very bad.

Having defined the geometric alignment function for a pair of samples (see Eq. (3)), we use it to compute the geometric alignment score $\beta_{C_k}^{(t)}$ of a coalition of size k:

$$\beta_{C_k}^{(t)} = \frac{f(k)}{\binom{k}{2}} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} G(\mathbf{S}_i^{(t)}, \mathbf{S}_j^{(t)}), \tag{4}$$

where $\mathbf{S}_i^{(t)}$ and $\mathbf{S}_j^{(t)}$ are target projections corresponding to the samples with the highest template match scores for coalition members m_i, m_j, respectively. Regarding the function $f(k)$, we have $f(1) = 0$ and it is non-decreasing for $k = 2, 3, \ldots, N$. The $\beta_{C_k}^{(t)}$ is analogous to the average of the geometric alignment of all possible tracker pairs in the coalition. In general, as the size of the coalition k increases, the average of the geometric alignment function

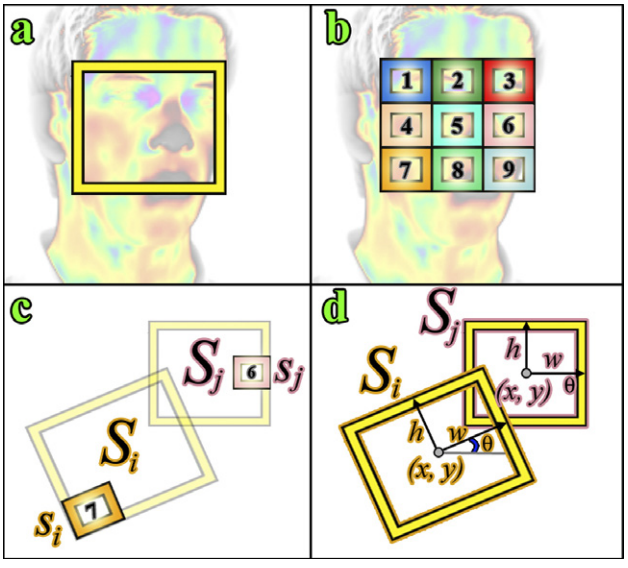

Fig. 3. Geometric alignment of tracker's target projections. (a) Target projection at $t = 0$. (b) Tracker network overlaid on the initial target projection. (c) Trackers 6 and 7 at a subsequent time t along with their corresponding target projections. (d) Parameterization of target projections to facilitate measurement of geometric alignment.

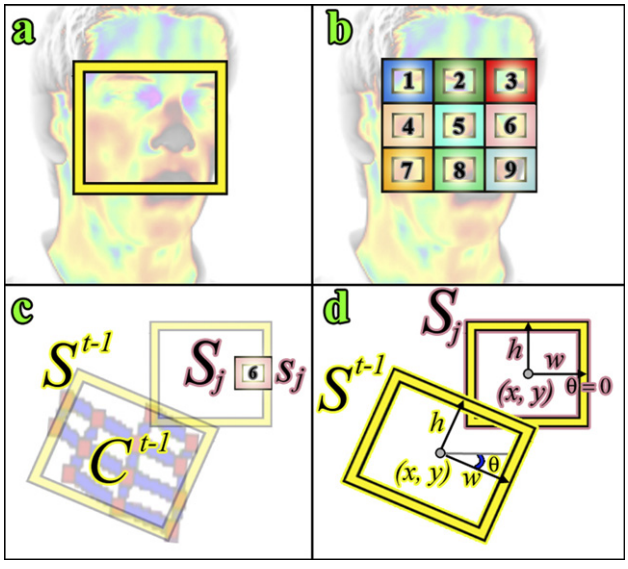

Fig. 4. Inter-frame projection agreement. (a) Target projection at $t = 0$. (b) Tracker network overlaid on the initial target projection. (c) The target projection at time $t - 1$ and tracker 6 with its corresponding target projection at time t. (d) Parameterization of target projections to facilitate measurement of inter-frame projection agreement.

of the members of the coalition decreases. To compensate for that loss, we introduced the linear function $f(k)$, whose role is to reward higher order coalitions as opposed to lower order ones.

We also use the geometric alignment function for a pair of samples (see Eq. (3)) to compute the inter-frame projection agreement $\gamma_{C_k}^{(t)}$ score (see Fig. 4):

$$\gamma_{C_k}^{(t)} = \frac{1}{k}\sum_{i=1}^{k} G(\mathbf{S}_i^{(t)}, \mathbf{S}^{(t-1)}), \tag{5}$$

where $\mathbf{S}_i^{(t)}$ is the target projection corresponding to the sample with the highest template match score for coalition member $m_i^{(t)}$ at time t; $\mathbf{S}^{(t-1)}$ is the target projection corresponding to the target state at time $t-1$ (previous frame).

The inter-frame membership retention score $\delta_{C_k}^{(t)}$ for a coalition C_k of size k at time t is given by:

$$\delta_{C_k}^{(t)} = \frac{1}{k}\sum_{i=1}^{k} \Delta\left(m_i^{(t)}, C^{t-1}\right) \tag{6}$$

where $m_i^{(t)}$ is the ith member of coalition C_k at time t, C^{t-1} is the winning coalition from the previous time step, and Δ is defined as:

$$\Delta(m, C) = \begin{cases} -1 & \text{if } m \text{ is not a member of } C, \\ +1 & \text{if } m \text{ is a member of } C, \end{cases} \tag{7}$$

where m is a tracker and C is a coalition.

Having defined the four scores, we proceed with the definition of the characteristic game function $\Pi^{(t)}(C_k)$:

$$\Pi^{(t)}(C_k) = \omega_\alpha \times \alpha_{C_k}^{(t)} + \omega_\beta \times \beta_{C_k}^{(t)} + \omega_\gamma \times \gamma_{C_k}^{(t)} + \omega_\delta \times \delta_{C_k}^{(t)}, \tag{8}$$

where ω_α, ω_β, ω_γ, and ω_δ are the weights (values range in $[0,1]$ and sum to 1) assigned to the four scores. Note that because of the function $f(k)$ in the geometric alignment score, the characteristic score may exceed the value of 1. This may happen in cases where we have higher order coalitions and quite good geometric alignment.

For every size of coalition $k \in \{1,2,\ldots,N\}$ we have $\binom{N}{k}$ different coalitions of size k, out of which we select the one with the highest payoff. To avoid complicating the symbology, let us continue to denote as C_k the preferred coalition of size k. Thus, we decide for the winning coalition C^t at time t to be:

$$C^t = \arg\max_{C_k} \Pi^{(t)}(C_k). \tag{9}$$

In coalitional game theory sometimes the characteristic function is a non-decreasing function of the size of the coalition (i.e., super-additivity) [28]. In our case this is not desirable because there may exist trackers that have lost their targets. In other words, the grand coalition (i.e., the coalition where all players/trackers participate) is not always the optimal to use. Thus, we need to give rewards to coalitions in such a way that the winning coalition is the coalition whose members best approximate the target. This is achieved by reducing the characteristic function of the coalition if it acquires poor trackers. Super-additivity is also related to the $f(k)$ function since if $f(k)$ increases, say exponentially, then the geometric alignment score $\beta_{C_k}^{(t)}$ will dominate the other three scores allowing super-additivity. In our case, having a linear $f(k)$ worked fairly well.

2.3. Target state estimation

We compute the final target state $\mathbf{S}^t$ from the winning coalition C^t in two steps. In the first step, we compute the deformation mesh M^t from the winning coalition. The deformation mesh M^t is composed of a set of points $\mathbf{A} = (\mathbf{a}_1, \ldots, \mathbf{a}_m)$, which are distributed over the selected target region during the initialization step. Each point is linked to anywhere between 1 and 4 trackers depending on its spatial location; 1 on the corners, 2 on the borders, and 4 on the inside. For each point, a transformation matrix $\mathbf{T}_{\mathbf{a}_i}^{\mathbf{c}_j}$ is computed that when applied to the center $\mathbf{c}_j$ of tracker j, gives the location of the point $\mathbf{a}_i$:

$$\mathbf{a}_i = \frac{1}{\sum_{j=1}^{n_i}\omega_j}\sum_{j=1}^{n_i}\begin{bmatrix} c_{jx} \\ c_{jy} \end{bmatrix}\mathbf{T}_{\mathbf{a}_i}^{\mathbf{c}_j}\omega_j, \tag{10}$$

where $\mathbf{a}_i$ is one of the points in the deformation mesh M^t, n_i is the number of trackers linked to the mesh point $\mathbf{a}_i$, and ω_j is the weight associated with tracker j. If the tracker j is a member of the winning coalition, the associated weight is the tracker's template match score ($\omega_j = \alpha_{m_j}^{(t)}$); otherwise it is 0 ($\omega_j = 0$).

Next, the four border points outlining the target projection in the clockwise direction $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_4)$ are computed from the mesh points $\mathbf{A}$ (see Fig. 5).

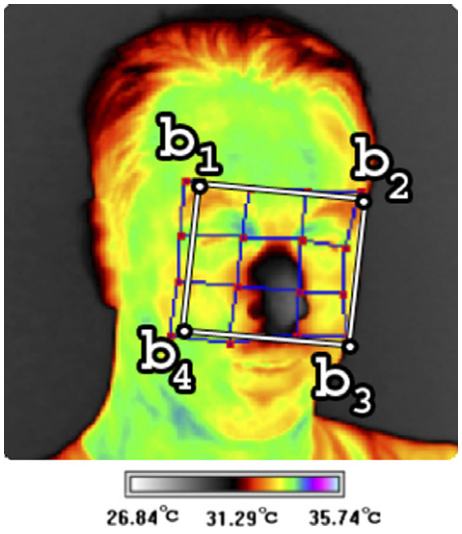

Fig. 5. Border points of the target projection. The target projection is shown in white. The deformation mesh is shown in blue and the deformation mesh points are shown in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

$$\mathbf{b}_i = \frac{1}{\omega_{\text{tot}}} \sum_{j=1}^{\text{m}} \begin{bmatrix} a_{jx} \\ a_{jy} \end{bmatrix} \mathbf{T}_{\mathbf{b}_i}^{\mathbf{a}_j} \omega_{a_j}, \tag{11}$$

where m is the number of points in the deformation mesh, $\mathbf{b}_i$ is one of the border points, $\mathbf{a}_j$ is one of the points in the deformation mesh M^t, $\mathbf{T}_{\mathbf{b}_i}^{\mathbf{a}_j}$ is the transformation from point $\mathbf{a}_j$ to $\mathbf{b}_i$, $\omega_{\mathbf{a}_j}$ is the weight associated with mesh point $\mathbf{a}_j$, which is the summation of each of its n_j member tracker weights:

$$\omega_{\mathbf{a}_j} = \sum_{k=1}^{n_j} \omega_k, \tag{12}$$

and ω_{tot} is the total weight of all mesh points α_j:

$$\omega_{\text{tot}} = \sum_{k=1}^{m} \omega_{\mathbf{a}_k}. \tag{13}$$

The second step is to compute the final target state $\mathbf{S}^t$ from the deformation mesh M^t by using the border points $\mathbf{B}$. The target parameter vector $\mathbf{P} = (p_1, \ldots, p_5)$, is defined as follows:

- p_1 is the x coordinate of the target center;
- p_2 is the y coordinate of the target center;
- p_3 is the rotation about the center of the target;
- p_4 is the width of the target;
- p_5 is the height of the target.

The parameter vector $\mathbf{P}$ is computed from the border points $\mathbf{B}$ of the winning coalition C^t as follows:

$$p_1 = \frac{1}{4} \sum_{i=1}^{4} b_{ix}, \tag{14}$$

$$p_2 = \frac{1}{4} \sum_{i=1}^{4} b_{iy}, \tag{15}$$

$$p_3 = \frac{1}{|C|} \sum_{i=1}^{|C|} c_{i\theta}, \tag{16}$$

$$p_4 = \frac{1}{2}\left((b_{1x} - b_{2x})^2 + (b_{1y} - b_{2y})^2\right)^{1/2} + \frac{1}{2}\left((b_{3x} - b_{4x})^2 + (b_{3y} - b_{4y})^2\right)^{1/2}, \tag{17}$$

$$p_5 = \frac{1}{2}\left((b_{1x} - b_{4x})^2 + (b_{1y} - b_{4y})^2\right)^{1/2} + \frac{1}{2}\left((b_{2x} - b_{3x})^2 + (b_{2y} - b_{3y})^2\right)^{1/2}, \tag{18}$$

where $|C|$ is the cardinality of the winning coalition.

2.4. Parameterization scheme

All particle filter trackers in the network are constructed the same way. Each tracker features 40 particles and performs a single iteration per frame.

The selection of the parameterization for the target is a compromise between complexity and speed. The parameter scheme excludes warping and 3D transformations and allows the remaining set of affine transformations. The tracking network helps to compensate for this simple parameterization, because each of the trackers in the network tracks a small portion of the target, and together they can approximate more complex transformations by the target (see Fig. 6).

A good analogy for our approach is using many simple geometric shapes, such as trapezoids, to approximate a complex area. Similarly, in our parameter selection we had to find a balance between a complex parameterization, which would more accurately describe the true range of target transformations, but would increase computational complexity exponentially, and a simpler representation, which would require more sub-division to indirectly approximate complex target transformations.

Another consideration is if we should use the direct parameter vector $\mathbf{P}$ (see Eqs. (14)–(18).) or a higher order derivative of it (see Fig. 7). To decide which parameterization model to use, we implemented both and performed an experiment using a simulated dataset in the thermal infrared (see Section 3.1 for more details). The implementation and comparative experimental results indicated that direct and derivative based modeling methodologies have their respective strengths and weaknesses. Direct parameter modeling, for example, allows for more global search in the target state space (see Fig. 7(c)). The downside is that direct modeling requires an explicit dynamics model. In contrast, derivative parameter modeling allows more detailed localized search (see Fig. 7(d)) but has poor

Fig. 6. Example of facial tracking in thermal video with a 3×3 tracking network. The columns show progressive frames as the subject looks left. The second row shows the individual trackers in the tracking network. The third row shows the deformation mesh. The fourth row shows the computed target state.

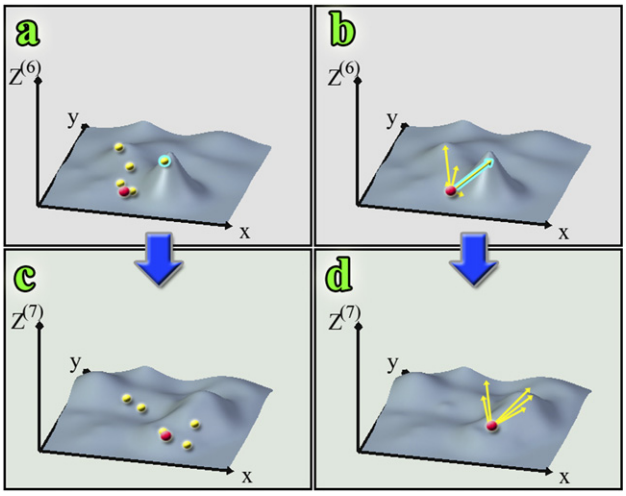

Fig. 7. Direct versus (first order) derivative parameter modeling propagation. The direct samples are shown as yellow points while the derivative samples are shown as yellow vectors. The previous state of the target $\mathbf{S}^{t-1}$, is shown as a red point. (a) and (b) show the direct and derivative samples, respectively, at frame t overlaid on top of the image observation $\mathbf{Z}^t$. The sample with the highest probability $\mathbf{s}^t$ is shown in cyan. (c) and (d) show the direct and derivative samples, respectively, at frame $t+1$ overlaid on top of the image observation $\mathbf{Z}^{t+1}$. The direct and derivative parameter models have been set to the same initial state in (a) and (b) and allowed to propagate freely in (c) and (d) to highlight the model differences. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

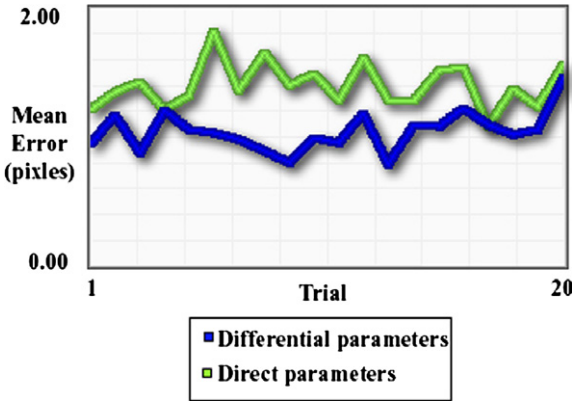

Fig. 8. Direct versus (first order) derivative parameter modeling for particle filter trackers. The performance curve of the directly parameterized tracker is shown in green while the derivative parameter one in blue. Both trackers were composed of a single particle filter tracker and were used to track the same target in 20 identical tracking trials involving target translation, scaling, and rotation. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

recovery capability if the target is temporarily lost. However, the increased accuracy that derivative parameter modeling offers in non-catastrophic cases (see Fig. 8) is very important to certain applications (e.g., physiological measurements in infrared). For this reason, we opted to use the first order derivative parameterization scheme.

Fig. 9. Recovery grid of trackers. (a) The target object is selected in the initial frame (blue rectangle). (b) The spawning positions for each of the recovery grid's trackers are shown in red. (c) The recovery grid trackers after 10 frames, with the tracker featuring the highest response sample drawn in white. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

2.5. Target recovery

To compensate for the poor recovery capability of our derivative parameterization scheme in the event of catastrophic loss of the target, we implemented an ancillary reset mechanism. Such a catastrophic loss of the target may happen due to extreme object deformations (e.g., a face turning away from the camera) or total occlusions. Our detection method uses a number of independent particle filter trackers that feature the direct parameterization model to facilitate global search in the target state space. The parameter set contains only the translation and rotation parameters for computational efficiency reasons. This is much simpler with respect to the parameter vector $\mathbf{P}$ (see Eqs. (14)–(18)) we use in the main game-based tracker.

The trackers are distributed uniformly in the image plane during initialization in the so-called spawning positions (see Fig. 9). Each tracker is then allowed to track for a specified number of frames before being reset to its spawning position. The recall interval is different for different trackers in the recovery grid. By recalling the trackers to their spawning positions in a staggered manner, we are trying to maximize coverage in the current 'hot' areas without allowing degeneration over time.

The state vector of the best sample from the recovery grid is used to influence the game-based tracking network.

2.6. Template matching

We opted to use a variant of the well-known technique of template matching [38], as the measurement vehicle for the particle filter trackers. Template matching is applicable to both visual and thermal band imagery [12]. The fundamental idea behind template matching is to create the template

and then to find the region in subsequent frames of video that most closely resembles the template. In our case, the template is a sub-sampling of pixels from the target object. The underlying assumption when using templates to represent the object of interest is that the appearance of the object will remain relatively constant throughout the course of the video. This limits the effectiveness of template matching in visual imaging under dynamic lighting conditions, unless more complex methods are employed to compensate, such as updating the template [39], or using model based light mapping [40]. Our main focus for this paper was tracking in thermal infrared video. Therefore, we opted not to implement one of these methods, and to simply constrain our visual band input to static lighting conditions.

In our framework we work with an input video function $\mathbf{V}(\mathbf{c},t)$, where $\mathbf{c}=(x,y)^T$ are pixel locations on the image plane and t is the time of the frame. In the case of thermal video we associate each pixel $\mathbf{c}=(x,y)^T$ with a temperature while in the case of visual video we assign red, green, and blue values to it. Therefore, the video function $\mathbf{V}(\mathbf{c},t)$ maps to a 3D matrix of temperatures in thermal infrared versus a 3D matrix of 3-tuples in visual imagery. Correspondingly, the template function $\mathbf{T}(\mathbf{c}^0)$ maps to a 2D matrix of temperatures in thermal infrared versus a 2D matrix of 3-tuples in visual imagery. The template function $\mathbf{T}(\mathbf{c}^0)$ is formed by sub-sampling the target region in the initial video frame $\mathbf{V}(\mathbf{c},0)$ (see Fig. 10).

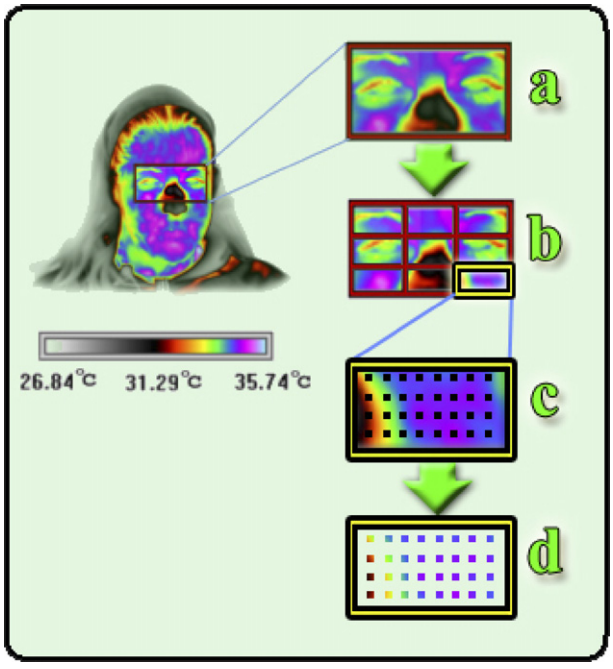

Fig. 10. Thermal template creation process. (a) The target object selected by the user. (b) The target selection is sub-divided into the individual trackers of the tracking network (c) The sub-sampling locations for the tracker outlined in yellow, are superimposed on the target object. (d) The temperatures extracted from the sub-sampling locations of the "yellow" tracker. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

The warp transformation matrix for the template pixels is $\mathbf{W}(\mathbf{p}^0;\mathbf{s},t)$,where $\mathbf{p}^0$ is the original parameter state vector $(t=0)$ and $\mathbf{s}=(s_x,s_y,s_\theta,s_w,s_h)^T$ is the parameter vector of sample $\mathbf{s}$ at time t for a tracker in the network (see Section 2.3). It is important to note that every tracker is associated with multiple samples (40 to be exact) at each time cycle, as part of its particle filter operation. We assume an approximately planar object and an affine warp transformation. The warp matrix $\mathbf{W}(\mathbf{p}^0;\mathbf{s},t)$ maps pixel $\mathbf{c}^0$ from the template $\mathbf{T}(\mathbf{c}^0)$ to a pixel $\mathbf{c}$ in video frame $\mathbf{V}(\mathbf{c},t)$:

$$\mathbf{c}=\mathbf{W}(\mathbf{p}^0;\mathbf{s},t)\times\mathbf{c}^0. \tag{19}$$

This produces a warped 2D temperature matrix $\mathbf{T}_\mathbf{s}(\mathbf{c})$, which is the projected template for the sample. Such warp transformations (Eq. (19)) are computed from all samples $\mathbf{s}$ of an individual tracker. To find the parameter vector of the optimal sample for the tracker we use:

$$\mathbf{p}^t=\arg\min_{\mathbf{s}}\sum_{\mathbf{c}}[\mathbf{T}_\mathbf{s}(\mathbf{c})-\mathbf{T}(\mathbf{c}^0)]^2. \tag{20}$$

The template match score of the tracker in question is computed then as:

$$\alpha_\mathbf{p}^t=\sum_{\mathbf{c}}[\mathbf{T}_\mathbf{p}(\mathbf{c})-\mathbf{T}(\mathbf{c}^0)]^2. \tag{21}$$

2.7. Optimization

We have used a profiler to find computational bottlenecks in our tracking algorithm. We have found that most of the computational time is spent in the template measurement function. Based on this finding, we have adopted the following optimization strategy:

1. We eliminate as many measurements as possible.
2. We parallelize the measurements that could not be eliminated.

For thermal tracking involving skin, we employ a simple skin detector to eliminate the need to measure samples that are not on the skin. We use a k-means clustering algorithm [41] to separate the thermal frame into two clusters, the 'hot' and 'cold' ones. We assume that the skin would belong to the hot cluster, an assumption that holds true for our dataset (faces in front of walls). The same technique could be applied in different scenarios with application specific selection of the number of clusters and target cluster.

Let us assume that we are interested in tracking the face of a human subject in thermal infrared for the purpose of measuring vital signs. A given frame at time t is composed of several columns of pixels. For each column we compute the mean temperature of all its pixels. We expect columns that cross the facial area to have elevated mean temperatures compared to background columns, which are typically cooler. By applying the k-means algorithm to

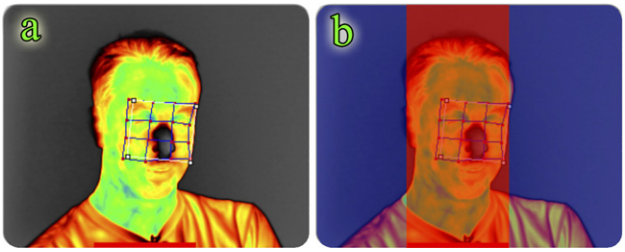

Fig. 11. Skin segmentation. (a) A typical thermal frame from the dataset. (b) Visualization of the hot and cold zones after the application of the k-means algorithm on the thermal frame.

these mean values we can cluster the columns into two zones: 'hot' and 'cold' (see Fig. 11). Then, all the samples of the individual trackers in the network that fall in a cold zone are automatically dropped from consideration.

The stochastic nature of the particle filter algorithm provides quite a few samples that can be easily ruled out from the reward calculations by the fast segmentation scheme described here. This significantly speeds up the trackers' performance. We could have implemented a skin color model [42] for facial tracking in the visual band to achieve a similar effect.

The distributed architecture of the tracking network lends itself particularly well to parallelization. Specifically, each measurement modifies only data local to the individual tracker while it performs read operations on the shared input image. Therefore, we can enhance performance by assigning each tracker to a separate working thread (multi-threading).

We gauged the effect of multi-threading on the performance of the tracking algorithm on a dual CPU computer with hyper-threading. Hyper-threading adds a virtual CPU for each physical processor, for a total of 4 processors. As can be seen in Fig. 12, the fully multi-threaded algorithm performed the fastest, with the most even distribution among the 4 processors. The result was an overall 32% extra processor usage for a gain of 4 frames per second, which represents a 33% increase in tracking speed. An intermediate performer was the partially multi-threaded algorithm, where we parallelized the selection and prediction (but not the measurement) steps of the individual particle filter trackers.

Finally, searching the full set of possible coalitions proved time consuming. Thus, we optimized the search by applying a greedy approach, which is linear to the number of trackers. Although, greedy search does not guarantee a global optimum, like a full search, it works well in practice. As the experimental results show (Section 4) it did not adversely affect the accuracy and robustness of coalitional tracking, while it reduced the time complexity by an order of magnitude. Specifically, the adopted search pruning algorithm works as follows:

(a) It selects the particle filter tracker with the best template match score.

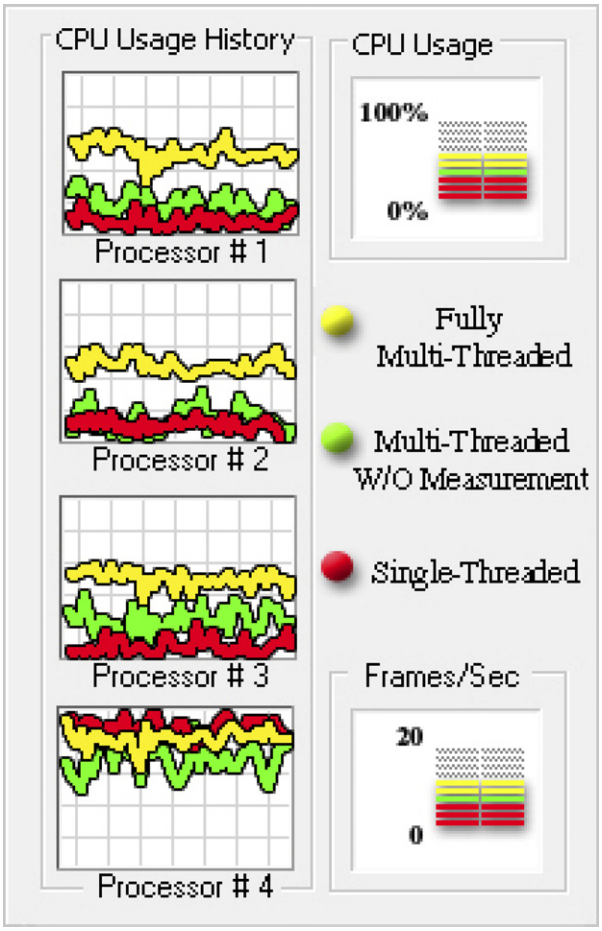

Fig. 12. Multi-threading performance enhancement. The experiment was first performed using a single-threaded version of the algorithm (Red). Then, the experiment was repeated with a version that multi-threaded everything except the measurement of the samples (Green). Finally, the experiment was run with a fully multi-threaded version (Yellow). The 4 processor usage curves for the three experiments are displayed on the left hand column. The right hand column shows both the percent CPU usage and the speed (frames/s) for each experiment. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

(b) For each of the remaining trackers in the network it takes its sample with the best template match score, pairs it with the sample from step #a, and computes the characteristic function (see Eq. (8)) of the, respectively, formed coalition.
(c) The best coalition from the previous step is then paired with each of the remaining trackers' best samples, and the characteristic function for each expanded coalition is computed.
(d) The algorithm repeats step #c until a coalition is formed, which contains as many members as there are trackers (grand coalition).
(e) The algorithm takes the best coalition of each size (number of members), and finds the global best among these. This is the winning coalition.

3. Experimental design

An important consideration in our experimental design was exact quantification of the tracker's performance. For this we needed an environment that would provide automatic ground-truthing. The cornerstone of our experimental design, however, was the provision to test our tracking method on video input from at least two different bands of the electromagnetic spectrum, one reflected and one radiated. The motivation was to demonstrate that the methodology is general enough to handle both. To satisfy this specification we performed experiments using visual band video (reflected) and mid-wave infrared video (radiated). The underlying implication being that if the tracker worked on both visual and mid-wave infrared video, then the tracker would be general enough to be adapted to other radiated bands, such as long-wave infrared, as well as other reflected bands, such as the near-infrared.

3.1. Design of simulated tracking environment

We used a simulated tracking environment to precisely quantify the tracker's performance. The environment was initialized to a frame of thermal video, and then the tracker was initialized to the target. The target to be tracked was then translated about the image plane while simultaneously undergoing transformations. Because the target transformations were dictated by the simulated environment, we could measure the true target state against its state projected by the tracker for each frame. Every simulated run was 200 frames in length.

3.2. Design of thermal infrared experiment

For the purpose of testing the tracking algorithm on thermal infrared video we selected a dataset that was used in previous publications [12]. It consists of 39 video clips, each containing a main human subject undergoing an interview. We chose to track 1000 frames of video from each of the subject clips, for a total of 39000 frames of video. The chosen video segments featured a temporary occlusion of the main subject by another subject who was passing through the field of view. More importantly, the clips featured out-of-plane rotation of facial tissue as subjects were rotating their heads left or right. We chose a single particle filter tracker [12] to compare against the coalitional tracker. Both the single particle filter and coalitional trackers featured identical parameterization. Both the single particle filter and the coalitional network were tasked to track exactly the same facial tissue of each subject. The ground-truthing of this experiment was the reconciliation of the observations of two independent operators.

3.3. Design of visual experiment

To demonstrate that the tracking methodology can also be applied to visual band video, we performed experiments on a series of visual videos, each containing a different type of target. These targets included faces and cans.

4. Experimental results

4.1. Results of simulated tracking environment

We first measured the accuracy of the coalitional tracker (see Fig. 13). The coalitional tracker maintained a mean error of about 1 pixel, which is sufficient for demanding applications, such as physiological measurements [5]. Moreover, the coalitional tracker exhibited consistent performance over 20 identical trials (see Fig. 13). This is extremely important because it would be impossible to extract useful physiological measurements if the tracker gave inconsistent results each time it was run. However, due to the stochastic nature of particle filtering, it is very difficult to altogether eliminate minute variability from the tracking result.

To determine the operational limits of the tracker we measured its error under increasingly faster target motion in the simulated tracking environment (see Fig. 14). The superior performance of the coalitional tracker in complex and fast transformations is evident.

The coalitional tracker was also capable of negotiating out-of-plane facial rotations (see Fig. 15) much more successfully than the single particle filter tracker (see Fig. 16).

4.2. Results of thermal infrared experiment

The results from the thermal infrared experiment (see Fig. 17 and Table 1) clearly show that the coalitional tracker provides superior tracking over the single particle

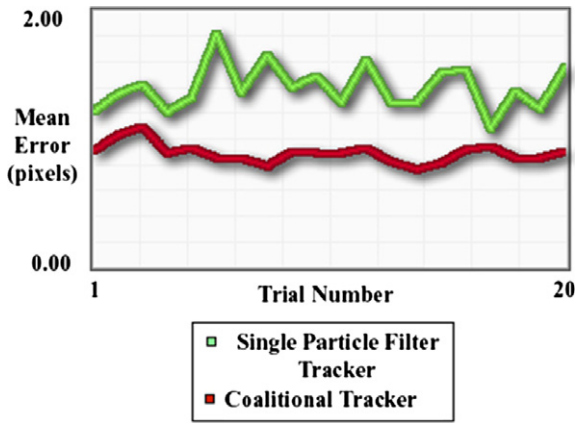

Fig. 13. Error and stability analysis of single particle filter (green) versus coalitional tracking (red). Both trackers were used to track the same target in 20 identical trials using the simulated tracking environment. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

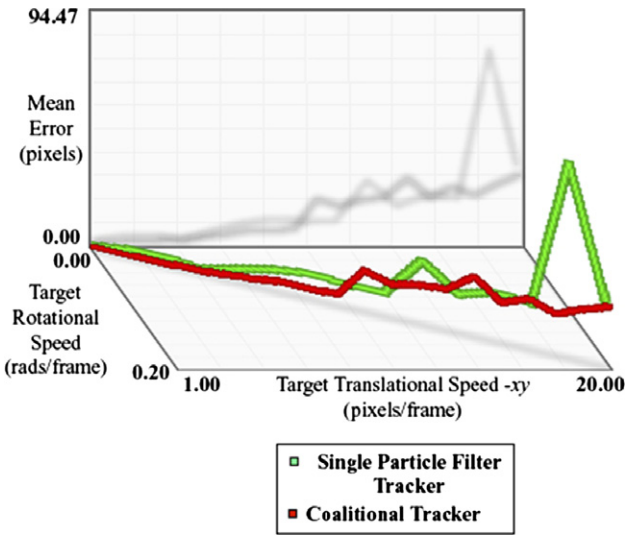

Fig. 14. Error analysis of single particle filter (green) versus coalitional tracking using the tracking network (red). Both trackers were used to track the same target in 20 trials using the simulated tracking environment. Each trial involved increasingly faster translational and rotational target motion. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

filter tracker. The proposed method proved robust in typical (see Fig. 20) and difficult (see Fig. 21) operational scenarios. The few failures of the coalitional tracker were mainly caused by significant out of plane rotation or substantial occlusion of the target (see Fig. 22).

A rare case of failure is exemplified in Fig. 18, when the subject experiences rapid physiological changes on a grand scale. The subject in the figure underwent facial temperature increase in excess of 2 °C within 6 minutes due to a state of high-anxiety.

This problem is due to the template measurement method, which assumes that the target's projection will not change dramatically over time. One possible solution to this problem is to dynamically update the template as presented in [39].

We extracted a sample physiological measurement from the subject in clip #2 and compared it against the respective

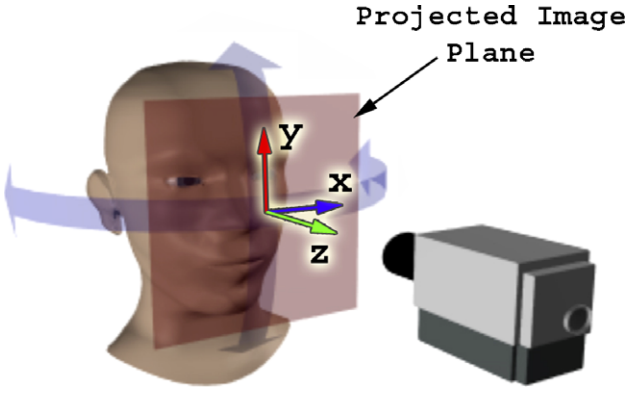

Fig. 15. Out-of-plane facial rotation. Any rotation which is not about the z-axis is considered out-of-plane rotation.

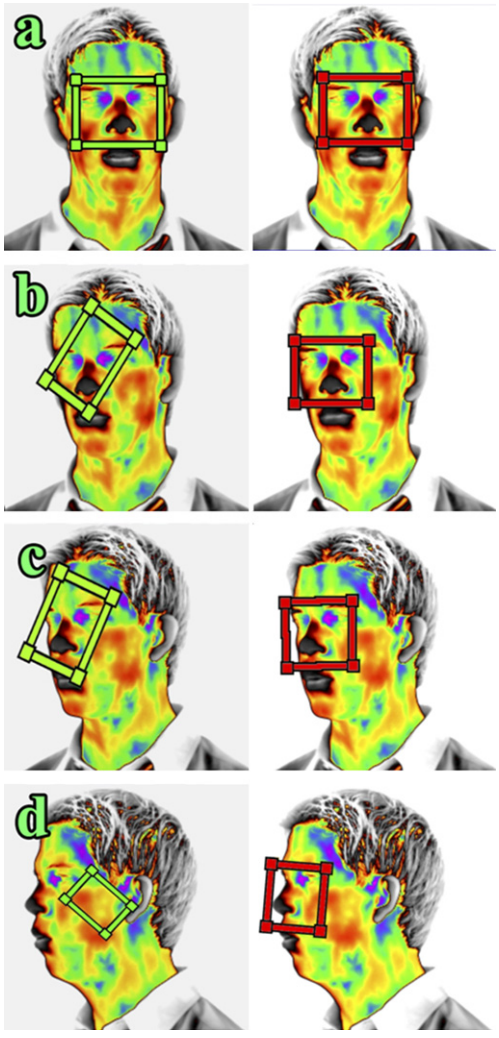

Fig. 16. Out-of-plane rotation comparison. The left column shows the single particle filter tracker (green) and the right column shows the coalitional tracker (red). (a) Initial frame. (b) and (c) Intermediate frames. (d) Final frame in a 1 min thermal clip. The poor performance of the single particle filter tracker is evident. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

ground-truth signal [12]. The measured signal is the mean temperature of the subject's periorbital area through the course of the video clip. It is evident that the coalitional tracker enables the acquisition of a signal nearly identical to the ground-truth (see Fig. 19), an indication of its fitness for accurate physiological measurements.

4.3. Results of visual experiment

The coalitional tracker performed robustly in several visual band experiments with various objects (faces and cans). The template was composed of 3-tuples (red, green, and blue reflectance values) instead of temperatures. The motion patterns included translation, rotation, and scaling (see Fig. 23 and Fig. 24).

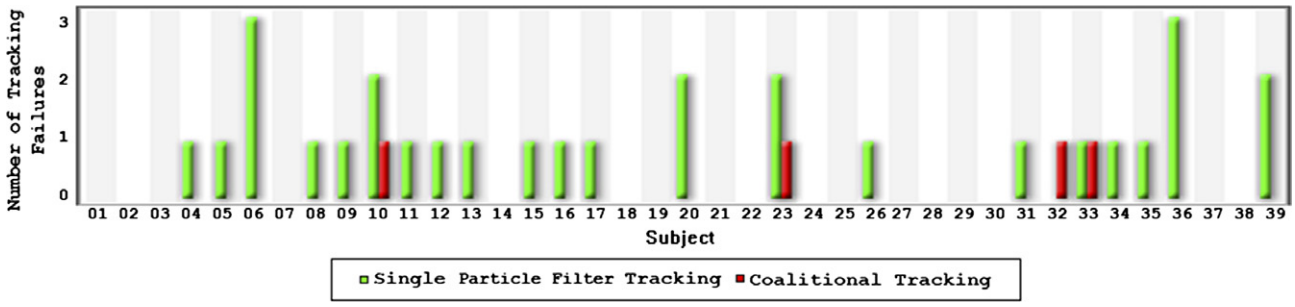

Fig. 17. Tracking failure graph for the 39 video clips in the thermal dataset. For each clip the number of single particle filter and coalitional tracking failures is shown in green and red, respectively. The absence of red bars in some video clip entries indicates perfect performance of the coalitional tracker. All 39 tracking videos can be viewed at http://www.cpl.uh.edu/html/users/jdowdall/html/Results/CVIU/results.htm.

Table 1
Causation of tracking failures in the thermal dataset

Reason for failure	Coalitional tracker failures	Single tracker failures
Target rotation	1	18
Partial occlusion	2	9
No recovery	1	2
Total	4	29

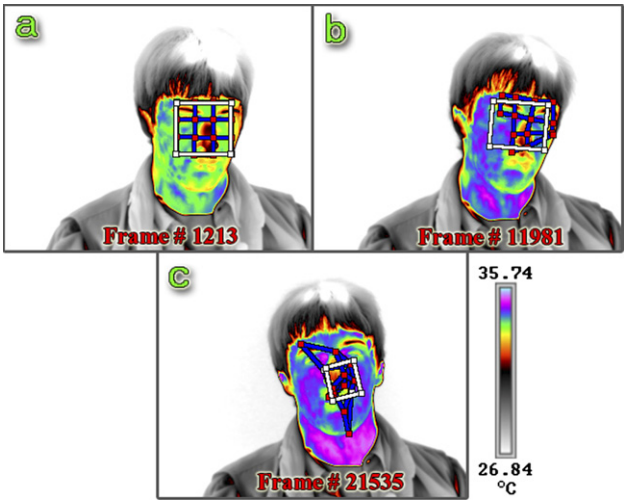

Fig. 18. Coalitional tracker performance under substantial physiological changes. (a) Tracker initialization. (b) The subject's face undergoes a substantial thermal change in the middle of the video clip. The tracker is still performing correctly, but the winning coalition is composed of fewer trackers that are able to follow their targets. (c) Towards the end of the clip, the subject's facial thermal profile continues to change dramatically and the coalitional tracker is off target.

5. Conclusion

We have proposed a novel tracking method. Our method uses a spatially distributed network of trackers whose interactions are modeled using coalitional game theory. The output of the method provides pixel level tracking accuracy, even in the presence of multi-dimensional target transformation.

We tested our method in thermal and visual video sets featuring faces and objects. We compared the performance

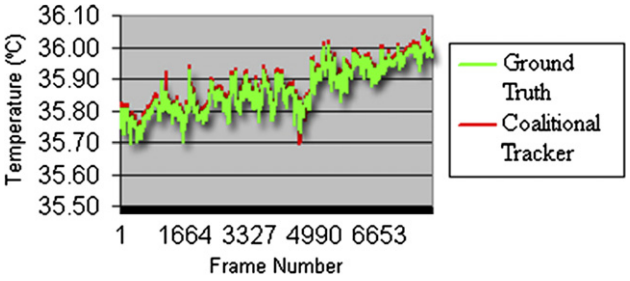

Fig. 19. Physiological signal extracted using coalitional tracking (in red) versus the ground-truth signal.

of the proposed coalitional tracker with that of a single particle filter tracker. The coalitional tracker exhibited superior performance in both regular and challenging tasks. The strength of the method comes from the redundancy that is elegantly encoded in its game theoretic structure. Detailed quantification and ground-truth verification indicate that the new method provides accuracy appropriate for demanding medical imaging applications. Equally important is the fact that the method appears to be general and flexible enough to be of use in imaging applications across the electromagnetic spectrum.

5.1. Future work

The particular adaptation of Game Theory to tracking presented in this paper is but one of many possible approaches that might be adopted. For example, the problem of tracking could be alternatively viewed as a non-cooperative game where the trackers compete with each other, and the final solution could then be modeled as a Nash (strategic) equilibrium [29]. Additionally, active research areas in Game Theory, such as Stochastic and Differential Games [43], could potentially be adapted for use in tracking.

An important area that is amenable to improvement is the current static template scheme. Although, it works well in the thermal infrared band, where emission of most objects does not change dramatically in short observation periods (e.g., a few minutes), it is potentially vulnerable in the visual band, where reflected light may change

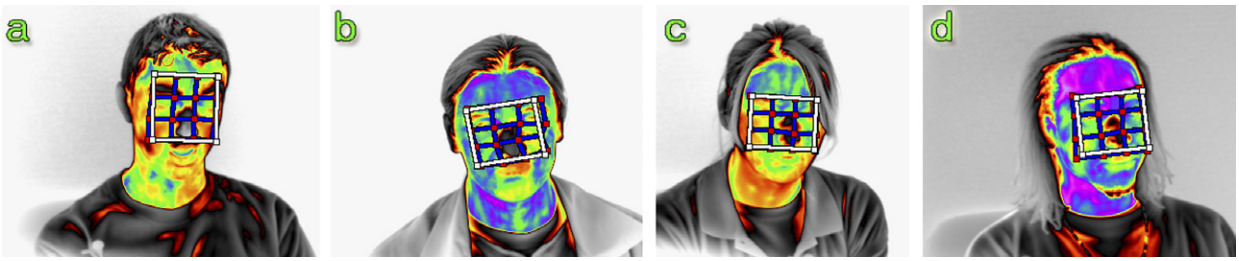

Fig. 20. Typical facial tracking examples from the thermal dataset. The selected subjects represent different ethnicities and both genders. (a) A Caucasian male. (b) A Caucasian female. (c) An Asian female. (d) A Black male.

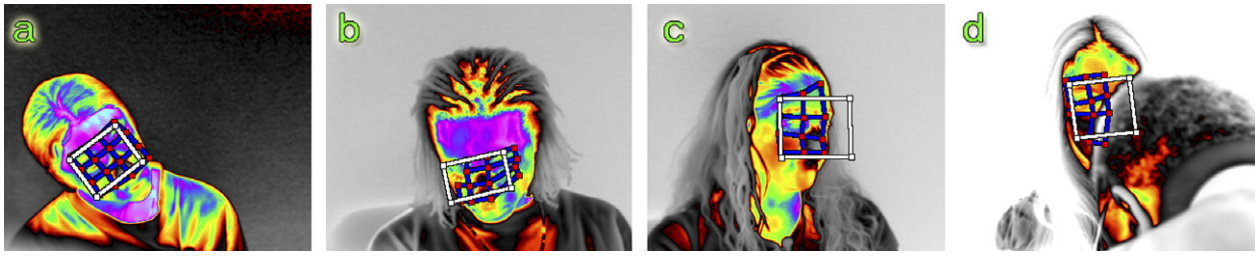

Fig. 21. Successful coalitional tracking in the presence of difficult circumstances in the thermal spectrum. (a) The target is rotating in-plane. (b) The target is rotating out-of-plane. (c) The target is rotating out-of-plane. (d) The target is partially occluded.

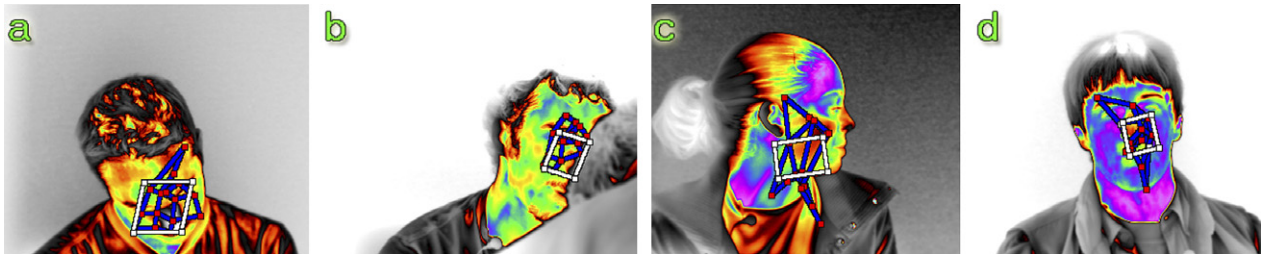

Fig. 22. Tracking failures in the thermal spectrum. (a) and (c) The target has rotated out of plane beyond the tracker's ability to compensate. (b) The original target (periorbital area) is largely occluded. (d) The target has undergone extreme physiological changes relative to the initial tracking frame (see Fig. 18 for more details).

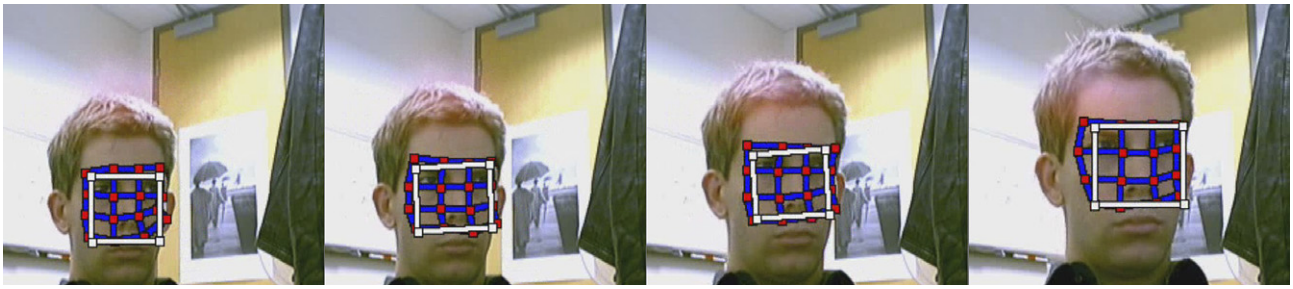

Fig. 23. Example of tracking a face experiencing scaling and translation in the visual spectrum. The frames are shown chronologically from left to right. The deformation mesh is shown in blue and the white rectangle represents the projected target state. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

dramatically over a split second depending on the angle of incidence. A dynamic updating mechanism for the template could potentially eliminate this vulnerability.

The current method is based on deterministic management of probabilistic trackers. A future method could be developed based on probabilistic management of probabilistic trackers. This could be realized within a Bayesian framework where the posterior weight of each tracker in the coalitional game would be computed from its prior and an appropriate likelihood function. This would add probabilistic memory into coalition membership. Therefore, one could eliminate the membership retention factor in the current characteristic function, which in essence crudely plays the same role.

Fig. 24. Example of tracking an object experiencing scaling and translation in the visual spectrum. The frames are shown chronologically from left to right. The deformation mesh is shown in blue and the white rectangle represents the projected target state. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

Acknowledgments

We thank the National Science Foundation (Grant No. IIS-0414754) and Dr. Ephraim Glinert as well as Dr. Andrew Ryan and the Department of Defense (multiple research contracts), for their support and encouragement during the lifetime of this burgeoning research project. The views expressed by the authors in this paper do not necessarily reflect the views of the funding agencies.

References

[1] D.A. Gonzalez, F.J. Madruga, M.A. Quintela, J.M. Lopez-Higuera, Defect assessment on radial heaters using infrared thermography, NDT & E International 38 (6) (2005) 428–432.
[2] M. Burrell, Computer vision for high-speed, high-volume manufacturing, in: Proceedings of the 1993 International Conference on Systems, Man, and Cybernetics, vol. 3, October 17–20, 1993, pp. 349–354.
[3] I. Pavlidis, V. Morellas, P. Tsiamyrtzis, S. Harp, Urban surveillance systems: from the laboratory to the commercial world, Proceedings of the IEEE 89 (10) (2001) 1478–1497.
[4] R.T. Collins, A.J. Lipton, H. Fujiyoshi, T. Kanade, Algorithms for cooperative multi-sensor surveillance, Proceedings of the IEEE 89 (10) (2001) 1456–1477.
[5] M. Garbey, A. Merla, I. Pavlidis, Estimation of blood flow speed and vessel location from thermal video, in: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, June 27–July 2, 2004, pp. 356–363
[6] N. Sun, M. Garbey, A. Merla, I. Pavlidis, Imaging the cardiovascular pulse, in: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, June 20–25, 2005, pp. 416–421.
[7] J. Fei, Z. Zhu, I. Pavlidis, Imaging breathing rate in the CO_2 absorption band, in: Proceedings of the 27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, September 1–4, 2005.
[8] J. Levine, I. Pavlidis, M. Cooper, The face of fear, The Lancet 357 (9270) (2001).
[9] I. Pavlidis, N.L. Eberhardt, J. Levine, Human behavior: seeing through the face of deception, Nature 415 (6867) (2002) 35.
[10] I. Pavlidis, J. Levine, Thermal image analysis for polygraph testing, IEEE Engineering in Medicine and Biology Magazine 21 (6) (2002) 56–64.
[11] C. Eveland, D. Socolinsky, L. Wolff, Tracking human faces in infrared video, Image and Vision Computing 21 (2003) 578–590.
[12] P. Tsiamyrtzis, J. Dowdall, D. Shastri, I. Pavlidis, M.G. Frank, P. Ekman, Lie detection—recovery of the periorbital signal through tandem tracking and noise suppression in thermal facial video, in: E.M. Carapezza (Ed.), Proceedings of SPIE Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense IV, vol. 5778, March 29–31, 2005.
[13] S. Krotosky, S. Cheng, M. Trivedi, Face detection and head tracking using stereo and thermal infrared cameras for "smart" airbags: a comparative analysis, in: Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems, vol. 1, 2004, pp. 17–22.
[14] A. Doucet, N. DeFreitas, N. Gordon (Eds.), Sequential Monte Carlo Methods in Practice, Springer-Verlag, 2001.
[15] M. Isard, A. Blake, Condensation—conditional density propagation for visual tracking, International Journal of Computer Vision 19 (1) (1998) 5–28.
[16] M. Isard, A. Blake, ICONDENSATION: Unifying low-level and high-level tracking in a stochastic framework, in: Proceedings of the 5th European Conference on Computer Vision, vol. 1, June 2–6, 1998, pp. 893–908.
[17] J. MacCormick, M. Isard, Partitioned sampling, articulated objects, and interface-quality hand tracking, in: Proceedings of the 7th European Conference on Computer Vision, vol. 1843, 2000, pp. 3–19.
[18] Y. Zhong, A.K. Jain, M.P. Dubuisson-Jolly, Object tracking using deformable templates, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (5) (2000) 544–549.
[19] Y. Shi, W. Karl, Real-time tracking using level sets, in: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, June 20–25, 2005, pp. 34–41.
[20] C. Zimmer, J.C. Olivo-Marin, Analyzing and capturing articulated hand motion in image sequences, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (11) (2005) 1838–1842.
[21] S. Goldenstein, C. Vogler, J. Stolfi, V. Pavlovic, D. Metaxas, Outlier rejection in deformable model tracking, in: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition, June 19–26, 2004.
[22] T.F. Cootes, G.J. Edwards, C.J. Taylor, Active appearance models, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (6) (2001) 681–685.
[23] F. Dornaika, J. Ahlberg, Efficient active appearance model for real-time head and facial feature tracking, in: Proceedings of the 2003 IEEE International Workshop on Analysis and Modeling of Faces and Gestures, October 13, 2003, pp. 173–180.
[24] C. Cheng, R. Ansari, A. Khokhar, Multiple object tracking with kernel particle filter, in: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, June 20–25, 2005, pp. 566–573.
[25] Y. Ting, W. Ying, Decentralized multiple target tracking using netted collaborative autonomous trackers, in: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, June 20–25, 2005, pp. 939–946.
[26] M. Isard, J. MacCormick, BraMBLe: A Bayesian multiple-blob tracker, in: Proceedings of the 8th IEEE International Conference on Computer Vision, vol. 2, July 7–14, 2001, pp. 34–41.

[27] J. MacCormick, A. Blake, A probabilistic exclusion principle for tracking multiple objects, International Journal of Computer Vision 39 (1) (2000) 57–71.
[28] T.S. Ferguson, Game Theory. Chapter IV, <http://www.math.ucla.edu/~tom/Game_Theory/Contents.html/>.
[29] K. Ritzberger, Foundations of Non-Cooperative Game Theory, Oxford University Press, Oxford, 2002.
[30] A. Rapoport, N-Person Game Theory: Concepts and Applications, Univeristy of Michigan, 1978.
[31] E. Rasmusen, Games & Information: An Introduction to Game Theory, Blackwell Publishers, Oxford, 1989.
[32] T.G. Fisher et al., Managerial Economics: A Game Theoretic Approach, Routledge, London, 2002.
[33] C. Schmidt (Ed.), Game Theory and Economic Analysis: A Quiet Revolution in Economics, Routledge, London, 2002.
[34] P. Ordeshook, Game Theory and Political Theory: An Introduction, Cambridge University Press, Cambridge, 1986.
[35] S. Brams, Game Theory and Politics, Free Press, New York, 1975.
[36] S. Hart (Ed.), Cooperation: Game-Theoretic Approaches, Springer-Verlag, Berlin, 1997.
[37] M. Mareš, Fuzzy Cooperative Games, Physica-Verlag, 2001.
[38] S. Baker, I. Matthews, Equivalence and efficiency of image alignment algorithms, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 2001, pp. 1090–1097.
[39] I. Matthews, T. Ishikawa, S. Baker, The template update problem, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (6) (2004) 810–815.
[40] Y. Adini, Y. Moses, S. Ullman, Face recognition: the problem of compensating for changes in illumination direction, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7) (1997) 721–732.
[41] J. MacQueen, Some methods for classification and analysis of multivariate observations, in: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, 1967, pp. 281–297.
[42] M. Jones, J. Rehg, Statistical color models with application to skin detection, in: Proceedings of the 1999 IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 1999, pp. 274–280.
[43] M. Bardi, T. Raghavan, T. Parthasarathy (Eds.), Stochastic and Differential Games: Theory and Numerical Methods, Annals of the International Society of Dynamic Games, Birkhaäser, 1998.