COST*: An Approach for Camera Selection and Multi-Object Inference Ordering in Dynamic Scenes

Abhinav Gupta
Dept. of Computer Science
University of Maryland
College Park, MD, USA
agupta@cs.umd.edu

Anurag Mittal
Dept. of Comp. Sc. and Engg.
IIT Madras
Chennai, India
amittal@cse.iitm.ernet.in

Larry S. Davis
Dept. of Computer Science
University of Maryland
College Park, MD, USA
lsd@cs.umd.edu

Abstract

Development of multiple camera based vision systems for analysis of dynamic objects such as humans is challenging due to occlusions and similarity in the appearance of a person with the background and other people- visual "confusion". Since occlusion and confusion depends on the presence of other people in the scene, it leads to a dependency structure where there are often loops in the resulting Bayesian network. While approaches such as loopy belief propagation can be used for inference, they are computationally expensive and convergence is not guaranteed in many situations.

We present a unified approach, COST, that reasons about such dependencies and yields an order for the inference of each person in a group of people and a set of cameras to be used for inferences for a person. Using the probabilistic distribution of the positions and appearances of people, COST performs visibility and confusion analysis for each part of each person and computes the amount of information that can be computed with and without more accurate estimation of the positions of other people. We present an optimization problem to select set of cameras and inference dependencies for each person which attempts to minimize the computational cost under given performance constraints. Results show the efficiency of COST in improving the performance of such systems and reducing the computational resources required.

1. Introduction

We consider the problem of multi-perspective analysis of moving people in crowded situations. Typical goals of such an analysis are to recover the position, orientation or the pose of each or some subset of the people in the scene. The analysis is difficult due to occlusions and appearance similarities of people with one another or the background against which they are viewed. We refer to errors arising from appearance similarities as "confusions". In multiple camera systems, information fusion needs to be sensitive to occlusions and confusions.

*Confusion and **O**cclusion analysis for **S**elections based on **T**asks

Our goal is to develop principled methods to "select" the camera(s) in which there is less occlusion and confusion for a particular person to infer that person's position or pose (See Figure 1). Additionally, we seek to identify the parts of the image where such occlusion and confusion occurs and use this information in the inference process. However, determining those regions of occlusion and confusion depends on the positions and poses of other people in the scene. This leads to a dependency structure for inference of position/pose of the people present in the scene, as is illustrated graphically in Figure 2(b). A Bayesian network for such multi-object inference will generally have loops. Those loops can be eliminated by appropriate selection of cameras and dropping inference dependencies which are not expected to yield significant information, as shown in the example in Figure 2(c).

We present COST, a framework to reason about such dependencies, that produces an inference order for multi-person, multi-perspective pose/position estimation. We additionally identify a set of cameras and the parts of the acquired images to be analyzed for each person. We show that COST not only yields a reduction in computational time compared to approaches such as Expectation Maximization (EM) or Loopy Belief Propagation (LBP) [16], but also shows quantitative improvement in the pose/position estimation due to camera selection.

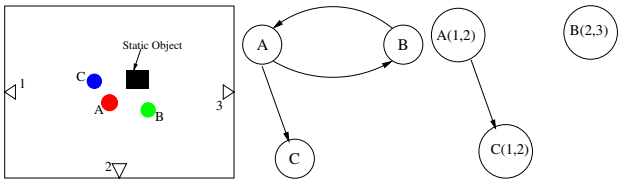
1.1. Related Work

There are many multi-perspective vision algorithms that analyze crowded scenes for either person position estimation or pose estimation. Most of the position estimation algorithms constrain the motion to a ground plane and perform inference by first segmenting the people in each view and then using data fusion techniques to obtain an estimate of the 3D locations of each person [15, 12, 13, 6]. While occlusion has been considered to some extent(for weighted fusion) in some papers [15, 13], confusion due to appearance similarities has not been previously considered. Additionally, most earlier work either ignores the inference dependencies or uses all of them, which makes the computation costly.

Previous work on pose estimation has only considered self-occlusion of one body part by another of the same per-

(1a) (1b) (2a) (2b) (3a) (3b)

Figure 1. Segmentation results and median line determination of a person in three different views. In view 1, there is no occlusion and confusion while in views 2 and 3 there is occlusion and confusion respectively. If the median lines are used for person position estimation as in [13, 10], without occlusion and confusion reasoning, we might mistakenly use the median lines shown in (2b) and (3b).



(a) Ground Plane (b) All cameras (c) COST

Figure 2. (a) A multiple-person scenario with 3 people and 3 cameras. (b)The dependency graph obtained if all cameras are used for estimation of all people. An edge $A \rightarrow B$ represents the information flow from A to B in the inference process-hence estimation of B depends on estimation of A. In this scenario, estimation of B depends on A due to occlusion in camera 1 and estimation of A and C depends on B and A respectively due to occlusions in cameras 3 and 2 respectively. (c) The dependency graph obtained if cameras are selected using COST. The selected cameras for estimation of each person are shown in the respective node. Since, camera 1 is not used for estimation of B , the estimation of B becomes independent of A. Additionally, if the degree of occlusion of C due to A is small (that is one cannot generate significant information for the estimation of C using the estimate of the location or pose of A) then one can also eliminate the dependency edge $A \rightarrow C$ without strongly affecting the accuracy of the result. Such elimination can be critical for loop removal when there are not enough cameras in which a person is isolated and discriminable.

son [9, 8, 19, 20]; occlusion of one person by another, leading to inference dependencies between people and their parts has not been addressed. A naive approach (by considering all pairwise interactions of all parts of all people) would involve constructing a large Bayesian network with loops; however, this results in an intractable optimization problem. We show how many of the loops in the Bayesian network can be eliminated using selection of the best cameras and the most important inference dependencies.

A related problem of sensor selection and information fusion has been studied in the field of sensor networks and distributed computing. The problem is to selectively choose the sensors so that information gain compensates for costs associated with information gathering. An optimal solution using such an information theoretic approach requires evaluating all possible combinations, making the problem NP-Hard. Denzler et. al [4] proposed an information theoretic based approach where the view which leads to maximum reduction in entropy is chosen. Since the computation of mutual information requires exponential time, other approximate [23] and heuristic based algorithms [22] have also been proposed. Other approaches in this field include use of look-up tables [17] or utility functions [2] in selection of camera views.

These information theoretic approaches only consider geometric analysis based on the fields of view of the cameras when computing mutual information. However, even though two cameras might have overlapping fields of view they can still provide different information due to occlusion and confusion. While [5] presents an approach for camera selection in the presence of occlusions, COST involves visibility and discriminability analysis in cojunction with reasoning about dependencies for camera selection.

Bayesian belief networks are an important mechanism for representation and reasoning under uncertainty. For a given belief-net even finding an approximate solution is NP-Hard [3]. Our approach is related to model simplification methods(see [7]), which simplify the model until exact methods become feasible. These approaches reduce the complexity by annihilating small probabilities [11] or removing weak dependencies [14] and arcs [21].

Our approach is complementary to these approaches. COST's loop removal procedure is primarily based on camera selection, which removes redundant and unreliable information in multi-perspective vision systems. Additionally, while previous approaches assume that weights of the dependencies are given, our approach considers occlusion and confusion in different cameras and removes loops based on this information.

The paper is organized as follows. We first describe how visibility and confusion factors for an object are computed in section 2. We then explain our optimization framework and a heuristic approach for fast approximate inference in

section 4. We finally present experimental results in section 5.

2. Computing Occlusion and Confusion

2.1. Computing Visibility

To estimate a property of a given person or object from a given camera, that person or object must be (partially) visible from that camera. But one person's visibility depends on the pose of other people in the scene, whose poses are generally known only probabilistically. This lends us to compute visibility probabilistically. Specifically, we compute the probability of visibility of each part of a person in each camera based on probabilistic estimates of the poses of all other people in the scene. To develop a generic formulation, let us consider an n-part model for a person where n is one for simple position estimation or ten for full body pose estimation.

Let dV be a differential volume element(voxel) which might be included in part j of person i. The Occluder Region, $\Omega^k(dV)$, of a differential element dV in camera k is defined as the 3D region in which another person, l, must be present so that dV would not be visible in camera k (See Fig 3). We also define the following events:

$E_{i,j}(dV)$ = Event that part j of person i includes dV [1]

$EO^k_{l,m}(dV)$ = Event that part m of person l intersects $\Omega^k(dV)$

$\overline{EO}^k(dV)$ = Event that no person intersects $\Omega^k(dV)$

The expected visibility of a part, that is, the number of visible voxels contained in that part, is then given by

$$E^v(i,j,k) = \int_{V^k} P(\overline{EO}^k(dV))P(E_{i,j}(dV))dV \qquad (1)$$

The probability that part m of person l does not occlude dV is the probability that part m does not contain any of the voxels that belongs to the set $\Omega^k(dV)$. Therefore, that probability is given by

$$P(\overline{EO}^k_{l,m}(dV)) = \prod_{dV_1 \in \Omega^k(dV)} 1 - P(E_{l,m}(dV_1)) \qquad (2)$$

The probability that no part of any person is in the occluder region is then given by [2]

$$P(\overline{EO}^k(dV)) = \prod_{(l,m)} P(\overline{EO}^k_{l,m}(dV)) \qquad (3)$$

Furthermore, in a tracking scenario, new people can enter the scene. In this case, we also need to consider the occlusions they are likely to introduce and how the expected visibility changes to account for new people. We assume there are a fixed and known number of locations, which we refer to as "portals", from which a new person enters or an existing person leaves the scene. Let $E_{new}(dV)$ be the event that a new person is present in voxel dV. The likelihood of this event, $P(E_{new}(dV))$, is the product of the likelihood that a portal is nearby (which is represented in terms of a prior probability $P^p(E_{new}(dV))$) and the image likelihood that a new person is seen in the region $P^L(E_{new}(dV))$. Therefore, $P(\overline{EO}^k(dV))$ is given by:

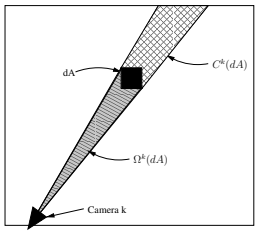


Figure 3. Schematic diagram showing $\Omega^k(dV)$ and $C^k(dV)$ projected on the ground-plane. Because of discretization, $\Omega^k(dV)$ and $C^k(dV)$ represent the set of voxels where another object must be present for occlusion or confusion to occur.

$$\prod_{dV_1 \in \Omega^k(dV)} (\prod_{(l,m)} 1 - P(E_{l,m}(dV_1)))(1 - P(E_{new}(dV_1))) \qquad (4)$$

2.2. Computing Confusion

Although a person(or some part) might be visible in view k , the view might still not be helpful in estimating the pose because of "camouflage" - his appearance being too similar to either the background or some other person(s) occluded by him. Due to such "confusion" with the "background", segmenting the person accurately would be problematic, and most pose inferences would degrade as the segmentation quality decreases.

Again, consider the differential element dV that a part (i,j) may contain. To compute the discriminability of dV, we determine the parts which can cause confusion. The confuser space $C^k(dV)$ of an element dV is defined as the region where the presence of a part (l,m) would cause confusion in the classification of a pixel that can be formed due to the projection of part (i,j) from dV (See Figure 3). The amount of confusion is proportional to the similarity in appearance of the two parts. We define the discriminability of a part (i,j) in a view k, $D^k(i,j)$ as:

$$D^k(i,j) = \sum_{(l,m)} c_{l,m} * d(a^k_{i,j}, a^k_{l,m}) + c_0 * d(a^k_{i,j}, B^k) \qquad (5)$$

where $a^k_{i,j}$ defines the appearance of a part (i,j), B^k defines the appearance of the background, d is a distance metric between the appearances and c is the corresponding weight. For example, if appearance is represented as a

[1] A part j can include many such voxels.

[2] By considering occlusion of a part (i,j) from itself, we implicitly select surface voxels instead of interior voxels. Interior voxels would be occluded the by surface voxels and would not be considered.

histogram, then d could be the dot-product of the two histograms or the earth mover's distance. The weight $c_{l,m}$ is proportional to the probability of the part (l, m) lying in the confuser space and being visible:

$$c_{l,m} = \frac{1}{Z} \int_{C^k(dV)} P(\overline{EO}^k(dA)) P(E_{l,m}(dV_1)) dV_1 \quad (6)$$

where Z is a normalizing factor. Hence, the expected number of discriminable voxels in view k contained in part (i, j) is given by:

$$I^k(i,j) = \int_V^k P(\overline{EO}^k(dV)) D^k(i,j) P(E_{i,j}(dV)) dV \quad (7)$$

3. Information in Views and Dependencies

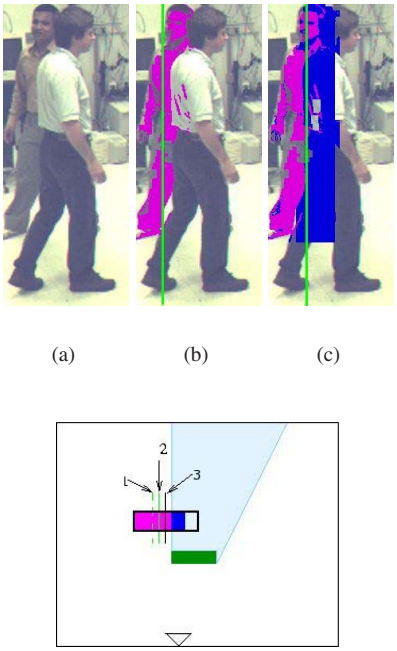
3.1. Model for Information Content

In order to perform inference reliably for some part of a given person using some view, that part should, ideally, not be occluded in that view and should not be "confused" with the background or other parts. The accuracy of the inference will depend upon both the degrees of occlusion and confusion, as discussed in the previous section. It will also depend on the uncertainty of such occlusion and confusion. We present a simple model for measuring the information available in a view regarding a part for the task of pose estimation. We say that a specific voxel belonging to a person is informative in some view if and only if it is both visible and discriminable. The information available about a specific part in a given view is then taken as the expected number of visible and discriminable voxels in that view.

3.2. Information from Dependencies

Inference decisions can be improved if estimates of the pose/appearance characteristics of the occluders and confusers are used. Such information can be employed in a variety of ways; an example for the position estimation problem is shown in Figure 4. Here the inference of a person's position involves constructing a median line through the silhouette of the person, and computing that line's intersection with the ground plane using calibration information. Figure 4(b) shows the segmentation of the person constructed from the visible and discriminable voxels. However, the estimate of the median-line is inaccurate when only these voxels are used (see the magenta voxels on the ground plane and median line-1 based on these voxels). If we additionally use the position of the occluder we can identify occluded regions (See light blue region in Figure 4(d)). The segmentation in the occluded region is then based on position priors, which would yield a better estimate of the median line as shown in Figure 4(c).

The inference of a part's position depends on the information about the occluders and confusers; the more accurate our information about the occluder and confusers, the more accurate will be our estimate. Thus, accurate inference of a part's position depends upon the inference of occluders and confusers. Such dependencies can be represented in a dependency graph (See Fig 2). Using the pose of other people in the inference process can, however, lead to loops in the Bayesian network. Additionally, using information from dependencies might involve expensive computation. Our goal is to avoid introducing edges into the dependency graph which either do not have sufficient information or introduce loops in the Bayesian network. We do this as follows: For each possible occluder or confuser l, we associate a binary decision variable, $\nu^k_{i,l}$ which represents whether the knowledge about the pose of person l is to be used in the inference of the pose of person i from view k [3]. If there is no edge from node l (the node representing person l) to node i in the dependency graph, then $\forall k, \nu^k_{i,l} = 0$. Given some selection of edges to include in the dependency graph, the total amount of information, $\Delta I^k_{i,j}$, that an algorithm can extract in view k about a part (i, j) using the



(a) (b) (c)

(d)

Figure 4. Importance of using occlusion information before fusion: (a) The original image (b) Occlusion-unaware segmentation and object inference, (c) Occlusion-aware segmentation and inference (d)The ground plane situation of the scenario. The black boundary show the actual voxels contained in the person. In case b, only the magenta voxels are used for median line estimation(1). In case c, one uses combination of magenta and blue voxels for estimation of median line(2). However, the true median line is represented by (3).

[3] In our model, dependencies are between people and not parts; we use the estimate of person l to estimate the locations of all parts of person i

estimates of its dependencies can be determined. This, however, also depends on the accuracy in the estimates of the dependencies.

4. The Optimization Problem

Given the amount of information available(with and without dependencies) regarding each person in each camera, we estimate the binary decision variables $\mu_i^k, \nu_{i,l}^k$ which represent whether or not camera k will be used in the inference of person $i(\mu_i^k)$ and, if so, whether to use the estimate of the pose of person l when estimating the pose of person i (that is whether or not we should include the edge from nodes l to i in the Bayesian network). For instance, in Figure 2 the decision variables (μ_C^1, μ_C^2, $\nu_{C,A}^2$) will be set to $true$ for person C. We would like to minimize the computational cost while guaranteeing that the expected error in the estimate of the pose of person i is below η_i(termed a "performance constraint"). Thus, the optimization problem can be formulated as

$$\min_{\mu_i,\nu_i} \sum_i J_i(\mu_i,\nu_i) \qquad \text{such that,} \qquad e_i(\mu_i,\nu_i) \leq \eta_i \quad \forall i \qquad (8)$$

where e_i represents the expected error in the estimate of the pose of person i and J_i represents the cost of computing the estimate of the pose of person i. This model also supports attention-based surveillance when $\exists i$ s.t $\forall_{j \neq i}$, $\eta_i << \eta_j$. In such a case, most of the computational resources would be devoted to estimating the pose of a distinguished person.

The optimization problem stated above is *NP-Hard* and belongs to the class of subset selection problems [18]. While approaches such as simulated-annealing can be used for optimization, much faster heuristic approaches can be employed.

4.1. A Heuristic Based Optimization Approach

We present a heuristic-based, greedy algorithm for the optimization problem. We build the dependency graph G by adding nodes one by one to G. Each node represents a person and the set of cameras selected for estimating the pose of that person. The edges incident on a node represent the dependencies to be used in estimation (An edge $l \rightarrow i$ indicates that to estimate the pose of person i, the pose of person l is used).

At each iteration, we compute the minimum cost [4] of estimation of each person, i, by selecting the best possible settings of the decision variables(μ_i and ν_i). However, to avoid loops in G, we require that dependencies be selected from the set of nodes already present in G, and should not introduce loops in the Bayesian network. The person with the lowest cost of estimation is then added to G. In the next iteration, the cost of estimation is re-computed, since the newly introduced node can be now used as a dependency for the remaining people.

[4] If the performance constraint for any person cannot be satisfied we assume the cost of estimation to be ∞

The algorithm is illustrated in Figure 5. At iteration 1, the minimum costs of computation are B=2 (Using camera 1,2 and no dependency), A=∞ (A needs to use dependency on either B or C for the performance constraint to be satisfied; since the dependency graph at t=0 is null, A cannot use any dependency), C=∞ (C also needs to use the estimate of B for its performance constraint to be satisfied). At iteration 2, the computation costs become A=8 (Using cameras 1,2,3 and the dependency from B) and C = 3 (Using cameras 2,3 and the dependency from B). Hence C is added at iteration 2. At iteration 3, the new minimum computation cost for A=4 (Using cameras 2,3 and the dependency from C. The dependency from B is not included since the performance constraint of A is satisfied without it)

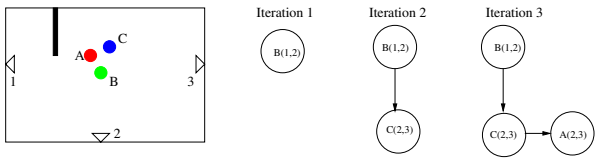


Figure 5. A sample scenario to illustrate the heuristic algorithm.

To compute the minimum cost of estimation for each remaining person at each iteration, one could exhaustively search the space of possible cameras and dependencies selection. However, such an approach requires exponential time in the number of cameras, so becomes infeasible when the number of cameras is large. Instead, we use a greedy approach, where we start by selecting a minimal set of cameras (two for example if pose is to estimated by stereo; one if position is estimated by intersecting a median line with the ground plane) and add more cameras and dependencies one at a time, based on the increase in cost of computation and the reduction in expected errors, until the performance constraints are satisfied.

5. Experiments

We next demonstrate how COST can be applied to multiple camera tracking algorithms.

5.1. Tracking People on a Ground Plane

5.1.1 Framework

We applied COST to a variant of M2Tracker. M2Tracker is a system that segments, detects and tracks multiple people on a ground plane in a cluttered scene [15]. The algorithm cycles between using segmentation to estimate people's ground plane positions and using ground plane position estimates to obtain segmentations; the process is iterated until stable. In M2Tracker, all people are segmented in all cameras; then the segmentations are combined using a wide-baseline stereo reconstruction algorithm for position estimation. In COST, selected people are segmented in selected views - those for which $\mu_i^k = 1$. To use the estimate of the position of an occluder, we first segment the occluder and then classify the pixels in the occluded region based on the prior probabilities alone.

M2Tracker uses a cylindrical model of the person with color encoded as a function of height. We use the same single part model. An estimate of the position probability densities on the ground plane is obtained using the estimates from the previous frame [5]. Given these probability densities, we can estimate the amount of visible information in each camera for each person. To compute the amount of information available through dependencies, we note that dependencies can be used to segment and classify pixels corresponding to occluded voxels. The number of occluded voxels that can be added due to dependencies depends on the selection of dependencies and the accuracy of the position estimate of the occluder. If the accuracy is higher, then we will be able to estimate the occluded region better, and, in turn, can better estimate occluded voxels. Therefore, the amount of information that can be extracted using occlusion is given by:

$$\Delta I_i^k = \int_{V^k} (1 - \prod_l P(\overline{EO}_l^k(dV))^{\nu_{i,l}^k(1-e_l)}) P(E_i(dV))dV \tag{9}$$

Based on the expected number of voxels which are visible and discriminable(I_i^k) and the expected number of occluded voxels that can be segmented using occlusion dependencies (ΔI_i^k), we compute the segmentation quality of each person in each view. Let S_i^k represent the segmentation quality of person i in view k; we compute S_i^k as :

$$S_i^k = I_i^k + \tilde{\lambda}\Delta I_i^k \tag{10}$$

where $\tilde{\lambda}$ defines the weight of occluded voxels as compared to visible and discriminable voxels.

We still need to define the error function, $e_i(\mu_i, \nu_i)$, which is the expected error in the estimate of the position of a person i for a given setting of the decision variables. This error depends upon

- **Segmentation Quality** The error in estimation decreases as the segmentation quality becomes better.
- **Camera Configuration:** The error due to fusion of information from different cameras not only depends on the segmentation quality in each camera but also on the configuration of the cameras. For example, in stereo reconstruction, performance depends upon the baseline. Thus, certain camera pairs would be preferred over others.

To define this error function, we observe that M2Tracker combines information from two cameras using stereo reconstruction to obtain a ground plane estimate. For a given camera pair, the error in estimation of position would increase as the segmentation quality decreases in either of the cameras in the pair. So, the error in estimating the position of person i using the stereo pair $(k1, k2)$ is approximated by

[5] In M2Tracker, visibility does not vary with height and hence ground plane analysis of visibility can be performed instead of 3D modeling

$$\mathcal{E}_i(k1, k2) = (1 - \tilde{f}(\theta_{k1,k2}) S_i^{k1} S_i^{k2}) \tag{11}$$

where $\theta_{k1,k2}$ is the angle between the viewing directions of cameras $k1$ and $k2$ on the ground plane. $\tilde{f}(\theta)$ represents how the accuracy of wide-baseline stereo varies with the angle between the viewing directions.

Additionally, M2Tracker fuses many camera pairs to obtain people's ground plane position estimates by using a weighted average of the estimates from each camera pair. Therefore, the error in the final position estimate would be the mean of the errors from individual camera pairs.

$$e_i(\mu_i, \nu_i) = \frac{\sum_{(k1,k2)} \mu_i^{k1} \mu_i^{k2} \mathcal{E}_i(k1, k2)}{\sum_{(k1,k2)} \mu_i^{k1} \mu_i^{k2}} \tag{12}$$

To define the computational cost function J_i, we observe that to apply the inference procedure to any person, we need to segment the person in the selected views and, for each dependency being used, we also need to segment the person providing the dependency in the selected view. The segments are then combined to estimate the person's position on the ground plane. We assume, for simplicity, that the computational cost of segmentation and wide-baseline stereo is some constant and independent of view and imaging conditions. Hence, the total cost of computation is given by:

$$J_i(\mu_i, \nu_i) = \sum_k \mu_i^k (\tilde{j_1} + \nu_i^k \tilde{j_2}) + \sum_{(k1,k2)} \mu_i^{k1} \mu_i^{k2} \tilde{j_3} \tag{13}$$

where $\tilde{j}$'s are constants. Intuitively, the cost is proportional to the total number of views used independently, and the number of dependencies utilized.

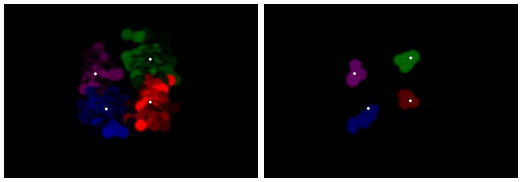
5.1.2 Results

We evaluated the performance of our implementation of M2Tracker with and without using COST on the publicly available dataset of M2Tracker. We used two sequences, one with four and the other with five people (15 cameras were used to record the sequences). The results of M2Tracker with four and eight cameras at uniform intervals around the circumference of a room provide our benchmark results. Figure 6 shows the tracking result using COST at frame 30 from three different views. Figure 7 shows the ground plane distribution of each person using COST (COST selected 2 cameras out of 15 for each person) and compares it with tracking using eight uniformly placed cameras in the original M2Tracker. It can be seen that M2Tracker has higher variance in position estimates using the eight camera system than COST has choosing only the "best" camera pair per person. This is because in many views a person is either occluded or confused with the background and this leads to inaccurate segmentations and subsequent errors in stereo reconstruction.

COST was also compared with four and eight camera M2Tracker systems in terms of the mean error in position estimation on the ground plane. The positional ground truth

(a) (b) (c)

Figure 6. Tracking Results at Frame 30.



(a) 8 Cameras (b) COST

Figure 7. Ground Plane Tracking Results: The variance of an estimate obtained using COST is low as compared to the M2Tracker system.

values were obtained manually. Figure 8 shows the performance of COST and M2tracker with 4 and 8 cameras on the 5 person sequence. It can be seen that the performance of COST is comparable to the system with 8 cameras. However, COST only analyses 2.2 cameras per person. This leads to an improvement in the computational speed of the system. In terms of the number of missed detections, COST outperforms M2Tracker with either 4 and 8 cameras (See Figure 9).

Experimental results indicate that it is generally sufficient to analyse only a small number of judiciously chosen cameras to obtain accuracy and performance similar to a system uniformly employing a large number of cameras. COST selects those few cameras and dependencies based on confusion and occlusion analysis. COST naturally chooses views in which people are visually isolated and only introduces a dependency when necessary - typically only when no isolated views were available.

5.2. Using COST for Multiple People Pose

We also applied the COST algorithm for full body pose estimation of multiple people. We implemented a 3D pose estimation system using non-parametric belief propagation [19, 9]. These papers have considered the problem of self-occlusion, but not of one person by another. The importance of considering occlusion of one person by another is illustrated in figure 10.

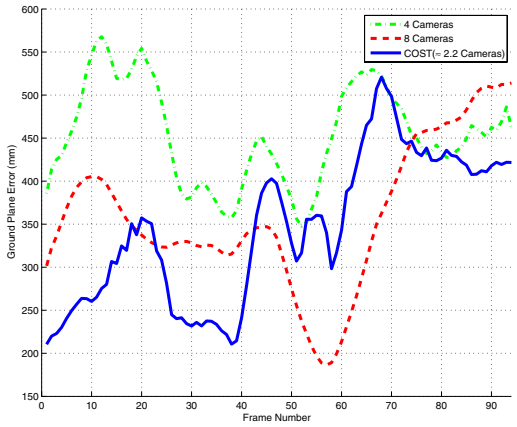


Figure 8. COST analysis Performance: Mean error in ground plane estimate in mm using COST analysis on M2Tracker. The system performs similarly to a system with 8 cameras. However, the cost of computation is substantially lower.

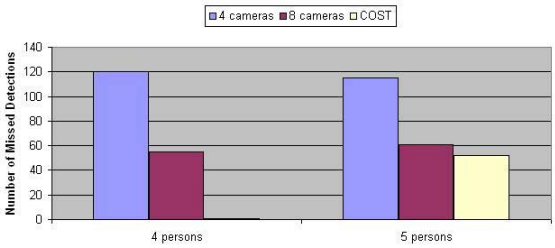
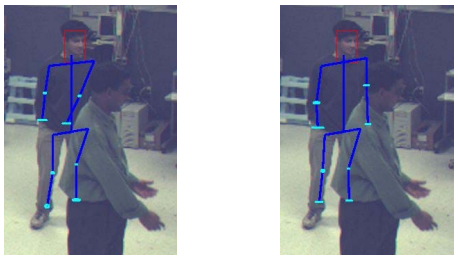


Figure 9. Number of Missed Detections: COST improves the system performance also in terms of missed detections. While none of the persons was missed using COST analysis in the 4 person sequence, the number of missed detections was also less than using 8 uniformly placed cameras in the 5 person sequence.

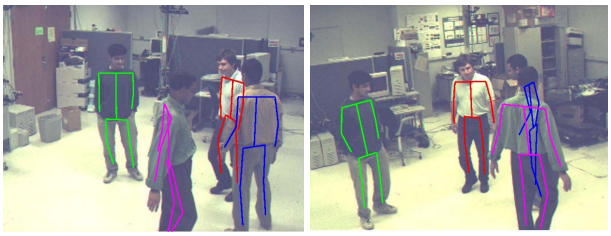
We used similar dependency and cost functions as for M2Tracker. However, the error function for full body pose problem was modified. As in [1], we compute the expected error in estimation of a person as the unweighted mean of



(a) Occlusion not considered (b) Occlusion is considered

Figure 10. Importance of considering occlusion information. The left hand is missed without considering occlusion information.

the expected errors in the estimation of the individual parts. Figure 11 shows an example of three views in which results of pose estimation are shown. COST uses the information from partially occluded views (which leads to a dependency) when there are no views available in which the person is visually isolated.



(a) (b)

Figure 11. Pose estimation by using COST loop removal.

6. Conclusion

We have presented a principled approach, COST, for camera and dependency selection for improving the performance and computational resource requirements for multi-camera systems. COST produces a directed acyclic dependency graph which can then be used to obtain an inference order using topological sort. The selection criteria in COST is based on visibility and "confusion" analysis in each view and the resulting dependencies. Experimental results indicate that COST outperforms a system which uses a large number of cameras for estimation of each person. Additionally, a COST based system is faster than other possible approaches based on EM and belief propagation which use all the cameras and dependencies for analysis.

7. Acknowledgement

This research was funded by the U.S. Government's VACE program. The authors would also like to thank Anuj Rawat for discussions on the work.

References

[1] A. Balan, L. Sigal, and M. Black. A quantitative evaluation of video-based 3d person tracking. In *VS-PETS*, 2005. 7

[2] D. Cook, P. Gmytrasiewicz, and L. Holder. Decision-theoretic cooperative sensor planning. *PAMI*, 1996. 2

[3] P. Dagum and M. Luby. Approximating probabilistic inference in bayesian belief networks. *AI*, 1993. 2

[4] J. Denzler and C. Brown. Information theoretic sensor data selection for active object recognition and state estimation. *PAMI*, 2002. 2

[5] A. Ercan, A. Gamal, and L. Guibas. Camera network node selection for target localization in the presence of occlusions. In *Workshop on DSC*, 2006. 2

[6] F. Fleuret, R. Lengagne, and P. Fua. Fixed point probability field for complex occlusion handling. In *ICCV*, 2005. 1

[7] H. Guo and W. Hsu. A survey of algorithms for real-time bayesian network inference. *Workshop on Real-Time Decision Support and Diagnosis Systems*, 2002. 2

[8] A. Gupta, A. Mittal, and L. Davis. Constraint integration for efficient multiview pose estimation with self-occlusions. *PAMI*. 2

[9] A. Gupta, A. Mittal, and L. Davis. Constraint integration for multi-view pose estimation of humans. In *3DPVT*, 2006. 2, 7

[10] W. Hu, M. Hu, X. Zhou, T. Tan, J. Lou, and S. Maybank. Principal axis-based correspondence between multiple cameras for people tracking. *PAMI*, 2002. 2

[11] F. Jensen and S. Andersen. Approximations in bayesian belief universes for knowledge-based systems. *Uncertainity in AI*, 1990. 2

[12] S. Khan and M. Shah. A multi-view approach to tracking people in crowded scenes using a planar homography constraints. In *ECCV*, 2006. 1

[13] K. Kim and L. Davis. Multi-camera tracking and segmentation of occluded people on ground plane using search-guided particle filtering. In *ECCV*, 2006. 1, 2

[14] U. Kjaerulff. Reduction of computation complexity in bayesian networks through removal of weak dependencies. *Uncertainity in AI*, 1994. 2

[15] A. Mittal and L. Davis. M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. *IJCV*, 2003. 1, 5

[16] K. Murphy, Y. Weiss, and M. Jordan. Loopy belief propagation for approximate inference: An empirical study. *Uncertainity in AI*, 1999. 1

[17] J. Park, P. Bhat, and A. Kak. A look-up table based approach for solving the camera selection problem in large area networks. In *Workshop on Distributed Smart Cameras*, 2006. 2

[18] K. Pruhs and G. Woeginger. Approximation schemes for a class of subset selection problems. In *LATIN*, 2004. 5

[19] L. Sigal and M. Black. Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. In *CVPR*, 2006. 2, 7

[20] E. Sudderth, M. Mandel, W. Freeman, and A. Wilsky. Distributed occlusion reasoning for tracking with nonparametric belief propagation. In *NIPS*, 2004. 2

[21] R. A. van Engelen. Approximating bayesian belief networks by arc removal. *PAMI*, 1997. 2

[22] H. Wang, K. Yao, G. Pottie, and D. Estrin. Entropy-based sensor selection heurestic for target localization. In *IPSN*, 2004. 2

[23] Y. Zhang and Q. Ji. Sensor selection for active information fusion. In *AAAI*, 2005. 2