行政院國家科學委員會補助專題研究計畫☒ 成 果 報 告 □期中進度報告

主動式多攝影機視訊監控系統之研究

計畫類別：☒ 個別型計畫　□ 整合型計畫

計畫編號：NSC 97-2221-E-009-132-MY3

執行期間：　99 年 8 月 1 日至　100 年 7 月 31 日

計畫主持人：王聖智
共同主持人：
計畫參與人員： 黃敬群、戴玉書、陳柏翔、宋秉修

成果報告類型(依經費核定清單規定繳交)：□精簡報告　☒完整報告

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外，得立即公開查詢
□涉及專利或其他智慧財產權，□一年□二年後可公開查詢

執行單位：交通大學電子工程系

中　華　民　國　100　年　10　月　31 日

主動式多攝影機視訊監控系統之研究

Visual Surveillance System with Multiple Active Cameras

計畫編號：NSC 97-2221-E-009-132-MY3

執行期限：99年8月1日至100年7月31日

主持人：王聖智 (交通大學電子工程系教授)

計畫參與人員：黃敬群、戴玉書、陳柏翔、宋秉修(交通大學電子所研究生)

中文摘要

在本計畫中，我們提出以貝氏階層式結構為基礎的分析方法，讓視訊監控系統得以用一致的架構，同時分析影像內容以及推論空間中場景的資訊。在真實的場景中，為了實現一套穩健的視訊監控系統，往往會面臨許多挑戰，諸如物體間相互遮蔽、前景物體與背景物體外貌相似而產生的混淆、透視投影所造成的物體形變、陰影的變化、還有外在光線變化造成的影像變異。透過將空間場景適當的參數化，並同時依據場景模型和擷取到的影像資料來進行分析，系統將能更輕易地處理前面所提及的變異因素。在貝氏階層式架構中，我們透過階層式表示法將以像素特徵為基礎的資訊、以區域影像內容為基礎的資訊、與以物件特性為基礎的資訊，透過機率的方式進行有系統的整合，以支援影像內容的分析與場景資訊的推論。透過所提出的貝氏階層式架構，前面所提到的許多變異因素可以被有效地解決，除此之外，某些變異因素還可進一步變成有效的線索來協助三維場景資訊的推論。

基於這樣的階層性結構，我們提出一套應用於多台攝影機之多角度人臉偵測系統。此系統可根據多攝影機擷取的影像偵測出影像中的人臉位置，並得到在三維空間中人臉方向的鳥瞰圖。有別於以往的作法，我們並不在二維的影像中直接作搜尋與偵測、或是將這些在二維的影像中的偵測結果投影到三維空間中作結合；反之，我們的系統直接在三維空間中進行搜尋，並將三維空間投影回二維影像中加以比對處理，以判斷在這個三維空間中是否存在著某種方向的人臉。這樣的做法使得我們的系統可以有效地結合多攝影機中的二維影像資訊，並可避免以往因在二維影像中的錯誤偵測所導致的不明確資訊整合。

關鍵詞：影像標記、圖學模型、物件偵測、物件追蹤、影像切割。

Abstract

In this project, we present a Bayesian hierarchical framework (BHF) to simultaneously deal with 3-D scene modeling and image analysis in a unified manner. In practice, to develop a robust video surveillance system, many challenging issues need to be taken into account, such as occlusion effect, appearance ambiguity between foreground and background, perspective effect, shadow effect, and lighting variations. Here, we find a way to handle these challenging issues by modeling 3-D scene in a parametric form and by integrating scene model and image observation together in the inference process. In the proposed hierarchical framework, we systematically integrate pixel-level information, region-level information, and object-level information in a probabilistic way for the semantic inference of image content and 3-D scene status. Under this BHF framework, occlusion effect, appearance ambiguity,

perspective effect, shadow effect, and lighting variations can be well handled. Actually, in the BHF framework, occlusion effect, perspective effect, and shadow effect may even provide useful clues to support 3-D scene inference.

Based on this BHF framework, we further propose a multi-view face detection system, which is capable of detecting all targets' faces in the given images and is able to illustrate the bird-eye view direction of each face in the 3-D space in a multi-camera surveillance system. Unlike existing approaches, the proposed system does not directly detect targets over the 2-D image domain nor project the 2-D detection results back to the 3-D space for correspondence. Instead, our system searches for the targets over small cubes in the 3-D space. Each searched 3-D cube is projected onto the 2-D camera views to determine the existence and direction of human faces. This approach can help us to efficiently combine 2-D information from different camera views and to suppress the ambiguity caused by 2-D detection errors.

Keywords: Image labeling, Graphical models, Object Detection, Object Tracking, Image Segmentation

1. OVERVIEW

A. System Overview

The main goal of our multi-camera system is to detect, locate, correspond, and label multiple targets and their faces, especially for walking people within the zone. In our approach, we decouple the locating of targets from the analysis of inter-occlusion. The basic idea is to detect the candidate target locations in the first stage and then spend computations only over those candidate locations for inter-occlusion inference. This two-stage procedure may preserve the accuracy of target location without dramatically increasing the computational cost.

The proposed scheme is majorly developed upon our earlier works [1-2] with two significant modifications. First, to suppress the ghost effect caused by geometric ambiguity, the 3-D scene model in our framework is defined in a probabilistic manner, rather than the deterministic form in our previous work [2]. Second, instead of applying a fixed 3-D target model to all tracked targets, we propose a Bayesian hierarchical framework with an expectation-maximization mechanism to refine the 3-D target model for each individual target. With the modification and extension of our previous works, the new system can locate, correspond, and label multiple targets over a multi-camera surveillance system, with the capability of ghost suppression and target model refinement.

B. System Flow

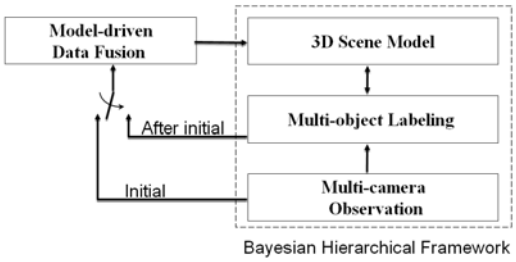


Figure 1. System flow of the proposed system.

In our fusion-inference scheme, we design a data fusion stage to detect candidate targets and their 3-D locations first. After that, in the inference stage, target identification, image labeling, and inter-occlusion are analyzed under the proposed Bayesian hierarchical framework (BHF) based on the fused 3-D priors. The inferred target labeling and correspondence results are further used to refine the 3-D target model, as illustrated in Figure 1.

In the data fusion stage, a model-based

approach is used to efficiently fuse consistent 2-D foreground detection results from multiple camera views. Here, we formulated a posterior distribution, named target detection probability (TDP), as the message pool to indicate the probability of having a moving target at a certain ground location. With the TDP distribution, candidate targets and their locations can be identified in a probabilistic manner. Moreover, with the use of 3-D target model, our fusion scheme may work well even with imperfect foreground extraction.

After data fusion, a set of candidate targets are detected, including both true targets and ghost targets. In our system, we use a few 3-D priors about the surveillance scenario, such as the probability distributions of target height and target location, to distinguish true targets from ghost targets. By properly integrating these priors into the scene knowledge, we can greatly simplify the ghost suppression problem. Moreover, in the BHF framework, we introduce a labeling layer as the interface between scene knowledge and multi-camera observations. This 3-layer framework unifies the target labeling, target correspondence, and ghost suppression into a Bayesian inference problem. Besides the intermediate role in the hierarchical framework, the labeling layer also provides a feedback route to refine the scene knowledge based on an EM (Expectation-Maximization) mechanism.

2. INFORMATION FUSION AND SUMMARIZATION

A. Foreground Detection on Single Camera

To fulfill the speed requirement of a real-time multi-camera system, we only consider 2-D foreground detection results as the observation data. For each camera, we build its reference background based on the Gaussian mixture model (GMM) approach [3]. To remove shadows, the frame difference operation is performed over the chromatic domain.

B. Information Fusion

To improve the accuracy in the estimation of target location, we adopt a model-driven approach to fuse 2-D information. In the proposed method, we define a Target Detection Probability (TDP) distribution to estimate the probability of having a moving target at a ground location, as expressed below:

$$G(X) \equiv p(X \mid F_1, \cdots, F_N; \Theta) \sim p(X)\, p(F_1, \cdots, F_N \mid X; \Theta) \quad (1)$$

In (1), X represents a location (x_1,x_2) on the ground plane of the 3-D space. N is the total number of cameras in the multi-camera system. F_i denotes the foreground detection result of the ith camera view. Θ defines the set of camera parameters of all N cameras.

To define F_i, we use (m,n) to represent the 2-D coordinate system of the ith camera. Based on the foreground detection result on the ith camera view, we define F_i as

$$F_i(m,n) = \begin{cases} 1 & \text{if } (m,n) \in V \text{ and } (m,n) \in \text{foreground regions} \\ 0 & \text{if } (m,n) \in V \text{ and } (m,n) \in \text{background regions} \\ P_L & \text{if } (m,n) \notin V \end{cases} \quad (2)$$

Moreover, given the location X, we assume the foreground detection results are conditionally independent of each other. With this assumption, we rewrite (1) as

$$p(X)\, p(F_1, \cdots, F_N \mid X) = p(X) \prod_{i=1}^{N} p(F_i \mid X) \quad (3)$$

To formulate $p(F_i|X)$, we model a moving person at the ground position X as a rectangular pillar, as shown in Figure 2. The height H and width R of the pillar are modeled as independent random

variables. Their prior probability $p(H)$ and $p(R)$ are assumed to be Gaussian and are pre-trained based on the data collected from the health center of our university. Based on the pre-calibrated projection matrix of the ith camera, a target at X with height H and width R is projected onto the image plane of the ith camera to obtain the projection region. Here we define the projection image M_i on the ith camera view as

$$M_i(m,n \mid H,R,X) = \begin{cases} 1 & if\ (m,n) \in projected\ regions \\ 0 & if\ (m,n) \notin projected\ regions \end{cases} \quad (4)$$

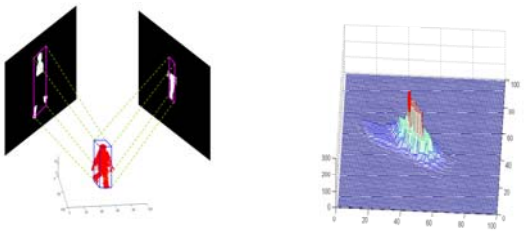


Figure 2. Proposed pillar model in the 3-D space and the estimated TDP distribution based on the foreground images.

With F_i and M_i, the normalized overlapping area Ω_i is defined as

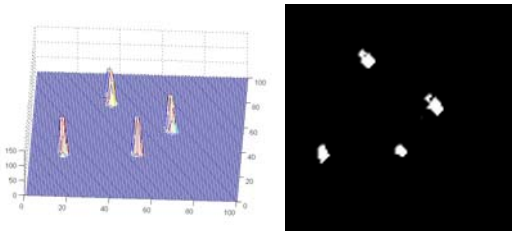
$$\Omega_i(H,R,X) \equiv \frac{\int\int F_i(m,n) M_i(m,n \mid H,R,X)\, dm\, dn}{\int\int M_i(m,n \mid H,R,X)\, dm\, dn} \quad (5)$$

By taking into account the prior probabilities $p(H)$ and $p(R)$, an estimate of $p(F_i|X)$ is defined as

$$p(F_i \mid X) \equiv \iint \Omega_i(H,R,X)\, p(H)\, p(R)\, dH dR \quad (6)$$

C. Information Summarization

Based on the TDP distribution, we obtain some useful information about the 3-D scene, including the number of candidate targets, the most likely position of each candidate target, and the unique ID of each candidate target. Typically, the TDP distribution contains several clusters, with each cluster indicating a moving target on the ground plane. Hence, the detection of multiple moving targets can be treated as a clustering problem over the TDP distribution.



(a) (b)

Figure 3. (a) TDP of four moving targets in the surveillance zone. (b)Bird-eye view of (a).

To perform clustering over the TDP distribution, we adopt the mean-shift clustering algorithm [4]. In the mean-shift algorithm, by iteratively calculating the next position y_{j+1} based on the following formula

$$y_{j+1} = \frac{\sum_{i=0}^{S-1} X_i W_i \exp\left(\left\| \frac{y_j - X_i}{h} \right\|^2\right)}{\sum_{i=0}^{S-1} W_i \exp\left(\left\| \frac{y_j - X_i}{h} \right\|^2\right)} \quad (7)$$

we can identify a few converging points. Those samples that converge to the same converging point are thought to belong to the same candidate target and are assigned the same ID. In (7), h is a parameter that controls the kernel size.

Assume we have identified M candidate targets on the ground plane with the ID's $\{T_1, T_2, \ldots, T_M\}$. If we denote the R_s samples that belong to T_k as $\{X_{k,0}, X_{k,1}, \ldots, X_{k,Rs-1}\}$ with the corresponding weights $\{W_{k,0}, W_{k,1}, \ldots, W_{k,Rs-1}\}$, we can estimate the position distribution function $p(X|T_k)$ for T_k. Here we model $p(X|T_k)$ as a Gaussian distribution. The mean vector and covariance matrix of $p(X|T_k)$ are estimated based on (8) and (9).

$$\mu^k = \left(\sum_{j=0}^{R_s-1} W_{k,j} X_{k,j}\right) \Big/ \left(\sum_{j=0}^{R_s-1} W_{k,j}\right) \quad (8)$$

$$\mathbf{C}^k = \left(\sum_{j=0}^{R_s-1} W_{k,j}(X_{k,j}-\mu^k)(X_{k,j}-\mu^k)^T\right) \Big/ \left(\sum_{j=0}^{R_s-1} W_{k,j}\right) \quad (9)$$

D. Ghost Object

From time to time, ghost clusters may occur in the TDP distribution. Geometrically, the ghost effect happens when the projection of a rectangular pillar at an incorrect location accidentally matches the foreground detection results on the camera views. In Figure 4, we present an illustration of the ghost problem when trying to reconstruct the 3-D scene based on two camera views.

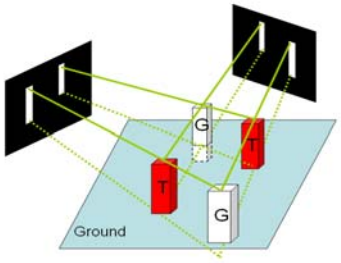


Figure 4. Illustration of the ghost problem.

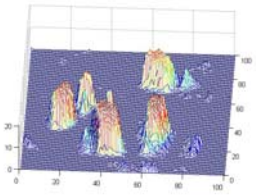
3. BAYESIAN INFERENCE AND GHOST SUPPRESSION

After information summarization, we identify a few candidate targets and their locations. For each candidate, we have to decide whether its status is "true" or "ghost". To determine the status of candidate targets, we consider not only foreground observations and geometric consistence but also some helpful prior knowledge about the targets.

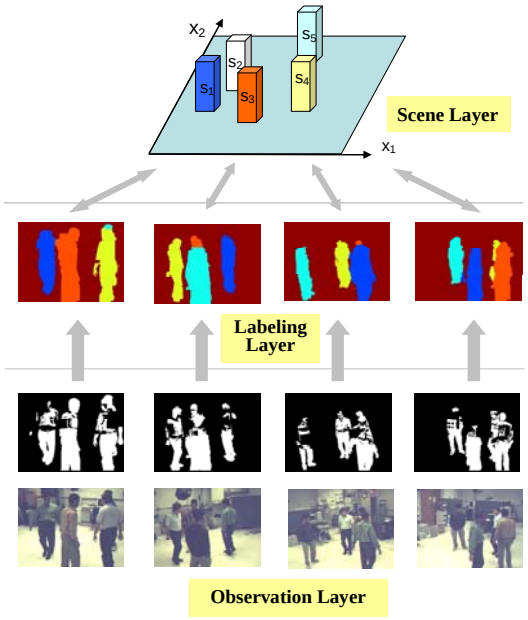
A. System Modeling

A.1. Bayesian Hierarchical Framework

Here, we propose a 3-layer Bayesian hierarchical framework (BHF) to simultaneously infer the status of candidate targets. In Figure 5, without loss of generality, we consider an example of TDP distribution fused from four camera views. The top layer of the BHF architecture is the scene layer S^L that indicates the 3-D scene knowledge built at the fusion stage. The bottom layer is the observation layer O^L, which contains both the original images and the foreground detection results. Here, we define $I_i(m,n)$ and $F_i(m,n)$ as the original image and the foreground detection result of the ith camera view. The value of $F_i(m,n)$ is defined as in Equation (2). Between the scene layer and the observation layer, a labeling layer L^L is inserted to deal with image labeling, target correspondence, and ghost removal. Here, we define $L_i(m,n)$ as the labeling image of the ith camera view.



(a)



(b)

Figure 5. (a) An example of TDP distribution fused from four camera views. (b) The corresponding Bayesian hierarchical framework.

A.2. Problem Formulation

In the "five candidate targets" case in Figure 6, the scene layer $S^L = \{s_1, s_2, s_3, s_4, s_5\}$ corresponds to the status of five candidate targets, with each status node being either true "1" or ghost "0". With five candidate targets, we have 2^5 status combinations in total. For each combination, we can generate the expected foreground occlusion pattern by approximating each "true" target as a rectangle pillar on the ground. By projecting the 3-D rectangle pillars onto each camera view, we form the expected foreground image. Ideally, the optimal status combination would lead to the best match between the expected foreground image and the detected foreground image. In Figure 6, we show two status combinations based on the example in Figure 5. By checking the projected foreground images, it appears that the latter combination is less likely than the former one.

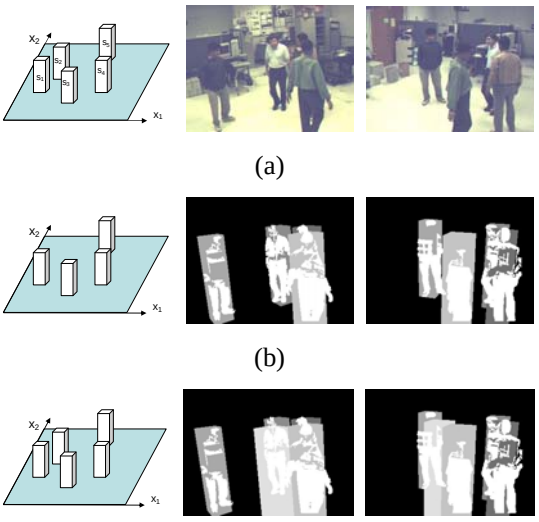


Figure 6. (a) The scene layer in Figure 5 and two of the four camera views. (b) The combination $\{s_1, s_2, s_3, s_4, s_5\}=\{1,0,1,1,1\}$ and the expected foreground images overlaid with the detected foreground images. (c) The combination $\{1,1,1,1,1\}$ and the expected foreground images overlaid with the detected foreground images.

If we denote I as the set of N original images, F as the set of N foreground detection images, L as the set of N labeling images, and S as a status combination, we unify the target labeling problem and the ghost suppression problem in a single MAP (Maximum A Posteriori) problem. In this problem, we seek the optimal status combination S^* and the optimal target labeling L^* that maximize the posteriori probability $p(L,S|I,F)$:

$$L^*, S^* = \arg\max_{L,S} p(L, S \mid I, F) \tag{10}$$

This equation is reformulated as below to decompose the inference problem into the combination of a few cross-layer issues in the BHF architecture

$$\begin{aligned} L^*, S^* &= \arg\max_{L,S} \ln p(L, S \mid I, F) \\ &= \arg\max_{L,S} \ln[\, p(I, F \mid L, S)\, p(L \mid S)\, p(S)] \\ &= \arg\max_{L,S} \ln[\, p(I, F \mid L)\, p(L \mid S)\, p(S)] \\ &= \arg\max_{L,S} [\ln p(I, F \mid L) + \ln P(L \mid S) + \ln p(S)] \end{aligned} \tag{11}$$

In (11), we assume $p(I,F|L,S) = p(I,F|L)$. That is, we assume the probabilistic property of the observed data I and F are independent of the status combination (S) once if the pixel labels (L) are determined. Besides, $\ln[p(I,F|L)]$ describes the relation between the labeling images and the observation data, $\ln[p(L|S)]$ describes the relation between the 3-D scene model and the 2-D labeling images, and $\ln[p(S)]$ describes the prior information about the 3-D scene model.

A.3. Formulation of p(I,F|L)

In our system, we formulate $p(I,F|L)$ as

$$p(I, F \mid L) = K \cdot \prod_i \prod_m \prod_n \exp(-E_D[F_i(m,n), L_i(m,n)]) \exp(-E_A[I_i(m,n), L_i(m,n); N_p]) \tag{12}$$

In (12), K is a normalization term. $E_D[F_i(m,n),L_i(m,n)]$ denotes the "detection energy" that relates the ith foreground detection image with the ith labeling image.

$E_A[I_i(m,n),L_i(m,n);N_p]$ denotes the "adjacency energy" that relates the ith original image with the ith labeling image by checking the adjacent property within the neighborhood N_p.

On the other hand, we define $E_D[F_i(m,n),L_i(m,n)]$ as

$$E_D(F_i(m,n),L_i(m,n)) \equiv \alpha \times \{1-\delta[F_i(m,n),T(L_i(m,n))]\} \tag{13}$$

with $T(L_i(m,n))$ being defined as

$$T(L_i(m,n)) = \begin{cases} 0 & \text{if } L_i(m,n) = T_0 \\ 1 & \text{otherwise} \end{cases} \tag{14}$$

and $\delta[p_a,q_a]$ being defined as

$$\delta[p_a,q_a] = \begin{cases} 1 & \text{if } p_a = q_a \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

In our system, by taking the original image $I_i(m,n)$ into consideration, we define the adjacency energy $E_A[I_i(m,n),L_i(m,n);N_p]$ based on a Markov random field to provide a smoothness constraint between adjacent labeling nodes [5]. Here, we define

$$\begin{aligned} &E_A[I_i(m,n),L_i(m,n);N_p] \\ &\equiv \beta \times \sum_{\Delta m=-p}^{p} \sum_{\Delta n=-p}^{p} C_A[I_i,L_i,m,n,\Delta m,\Delta n] \end{aligned} \tag{16}$$

where

$$\begin{aligned} &C_A[I_i,L_i,m,n,\Delta m,\Delta n] \\ &\equiv (1-\delta[L_i(m,n),L_i(m+\Delta m,n+\Delta n)]) \\ &\quad \times G_S(\|I_i(m,n)-I_i(m+\Delta m,n+\Delta n)\|) \end{aligned} \tag{17}$$

In (16), N_p denotes the $(2p+1)\times(2p+1)$ neighborhood around (m,n), and β is a learned penalty constant whose value is to be determined later. In (17), $\delta[p_a,q_a]$ is defined as in (15). In our system, we design $G_S(U)$ to be a discriminative function similar to a logistic sigmoid function:

$$G_S(U) = Sigm(U)+1 = (1-e^{\rho(U-C_{th})})/(1+e^{\rho(U-C_{th})})+1 \tag{18}$$

Here, $Sigm(U)$ outputs a positive value if U is smaller than C_{th}, and outputs a negative value otherwise. With this design, $C_A[.]$ is equal to zero when $L_i(m,n)$ and $L_i(m+\Delta m,n+\Delta n)$ are the same. If $L_i(m,n)$ and $L_i(m+\Delta m,n+\Delta n)$ are different, $C_A[.]$ gives a larger penalty if the difference between $I_i(m,n)$ and $I_i(m+\Delta m,n+\Delta n)$ is smaller than C_{th}. Hence, $L_i(m,n)$ and $L_i(m+\Delta m,n+\Delta n)$ tend to share the same label when the difference between $I_i(m,n)$ and $I_i(m+\Delta m,n+\Delta n)$ is small, and tend to have different labels otherwise.

Figure 7. Examples of $p(L_i(m,n) = T_k \mid S)$

A.4. Formulation of p(L|S)

Given a status combination S, we define a conditional probability $p(L_i(m,n)=T_k|\ S)$ to express the likelihood of having a label T_k at the pixel (m,n) of the ith labeling image. Here, with the status combination S, we define a few rectangular pillars on the ground. The height and width of each pillar are sampled from $p(H)$ and $p(R)$. The locations of the pillars are sampled from $p(X|T_k)$, where T_k indicates the kth target. With the camera projection parameters, the expected foreground patterns for each target can be generated by projecting these rectangular pillars onto each camera view. Occasionally, more than two targets may project onto the same image region and cause occlusion. The inter-occluded patterns can be determined by checking the distance from the camera to the mean location of the targets. In Figure 7, we demonstrate the occlusion effect by plotting $p(L_i(m,n)=T_k|\ S)$ individually for each of the four targets in Figure 5(b).

Based on the definition of $p(L_i(m,n)=T_k|\ S)$, we define the log probability function $\ln[p(L|S)]$ as

$$P(L \mid S) \equiv \prod_i \prod_m \prod_n p(L_i(m,n) \mid S) \qquad (19)$$

and thus

$$\ln P(L \mid S) = \sum_i \sum_m \sum_n \ln\, p(L_i(m,n) \mid S) \qquad (20)$$

In our system, we assume the number of true targets at the current moment would be similar to that at the previous time instant. Hence, if we denote S_o^{t-1} as the optimal status combination at the previous time instant (t-1) and S^t as a status combination at the current time instant t, we define the prior probability of S^t as

$$p(S^t) = \begin{cases} W_1, & \text{if } \left|N(S^t) - N(S_o^{t-1})\right| \le 1 \\ W_2, & \text{otherwise} \end{cases} \qquad (21)$$

where W_1 and W_2 are two constants with $W_1 \ge W_2$. In (21), $N(S)$ denotes the number of true targets in the status combination S. In detail, if we know the ratio between W_1 and W_2, we can determine the value of W_2 such that the probability summation equals to 1.

B. Multi-Target Labeling with Ghost Suppression

B.1. System Formulation

With the above deduction, the labeling of targets and the suppression of ghost targets can be solved by finding the optimal labeling images (L^*) and status combination (S^*) that maximize the following potential function $C_p(L,S)$:

$$\begin{aligned} L^*, S^* &= \arg\max_{L,S} C_p(L,S) \\ &= \arg\max_{L,S} \{ -\sum_i \sum_m \sum_n E_D[F_i(m,n), L_i(m,n)] \\ &\quad - \sum_i \sum_m \sum_n E_A[I_i(m,n), L_i(m,n); N_p] \\ &\quad + \sum_i \sum_m \sum_n \ln\, p(L_i(m,n) \mid S) + \ln\, p(S) \} \end{aligned} \qquad (22)$$

In (22), we incorporate detection energy E_D, adjacency energy E_A, likelihood function $p(L|S)$, and prior probability $p(S)$. As mentioned before, the detection energy $E_D(F_i(m,n),L_i(m,n))$ represents the bottom-up constraint between the foreground detection images and the labeling images. The likelihood function $p(L|S)$ represents the expected labeling layout based on the status combination S. The expected inter-occluded patterns among candidate targets are modeled in $p(L|S)$ to influence the classification of local labeling nodes. By introducing the adjacency energy $E_A[I_i(m,n),L_i(m,n);N_p]$, the proposed framework can not only infer the labeling based on the fusion of scene knowledge and foreground detection results, but also refine the labeling results based on the original image data. Last, the prior probability $p(S)$ includes the temporal prediction based on the previous decision.

B.2. System Formulation

In Equation (22), (α,β) control the weights of detection energy E_D and adjacency energy E_A in the potential function $C_p(L,S)$. To determine (α,β), the method proposed by Yu et al. [3] is adopted. In detail, With the ground truth of our training data, we can manually label the optimal solution (L^*,S^*) and the true target locations on the ground plane that maximize $C_p(L,S)$. For any other degraded solution (L^d,S^d), we have the relationship $C_p(L^*,S^*;\alpha,\beta) \ge C_p(L^d,S^d;\alpha,\beta)$ that leads to an inequality constraint for α and β. After having collected an enough number of constraints for α and β, the optimal parameter set (α^*,β^*) can be found by finding the maximal summation over the entire solution space of α and β subject to the collected constraints. That is,

$$\begin{aligned} &(\alpha^*, \beta^*) = \arg Max(\alpha + \beta) \\ &subject\ to:\ \alpha \ge 0\ and\ \beta \ge 0 \\ &subject\ to:\ C_p(L^*, S^*; \alpha, \beta) \ge C_p(L^{d-i}, S^{d-i}; \alpha, \beta)\,|_{i=1\sim T_n} \end{aligned} \qquad (23)$$

where T_n is the number of the degraded solutions used for training. The optimization problem in (23)

is then solved by using a Linear Programming method.

B.3. Optimal Status Inference and Target Labeling

For each status hypothesis S^H, we deduce the optimal L that maximizes the potential function $C_p(L,S=S^H)$ in (22). If we treat $E_D(F_i(m,n),L_i(m,n))$, $p(L_i(m,n)|S=S^H)$, and $p(S=S^H)$ as data terms and treat $E_A[I_i(m,n), L_i(m,n);N_p]$ as a smoothness term, $C_p(L,S=S^H)$ actually follows a canonical form that can be maximized based on many existing optimization algorithms [5-7]. Based on a recent study, the graph cuts method is proved to be more efficient in terms of running time as compared to loopy belief propagation, tree-reweighted, and iterated conditional mode algorithms [7]. Hence, in our system, the graph cuts algorithm [8-10] is used for the maximization of $C_p(L,S=S^H)$.

In our system, the optimal image labeling under S^H are achieved by assigning to each pixel a suitable ID from the set $\{T_0, T_1, \ldots, T_M\}$. Based on the graph cuts theory [8-10], we form a graph in Figure 8 to represent our optimization problem. In this graph, a candidate target, say T_1, can only affect a portion of labeling nodes in the labeling image. Through the projection of the 3-D candidate target onto the ith camera view, the relation is represented by a collection of "t-links". In our system, we combine $E_D(F_i(m,n),L_i(m,n))$, $p(L_i(m,n)|S=S^H)$, and the prior $p(S= S^H)$ as the data term to define the weight of each t-link. Moreover, the "n-links" in the graph represent the smoothness term, which is modeled as $E_A[I_i(m,n),L_i(m,n);N_p]$. After forming this graph, our optimization problem is equivalent to the cutting of the t-links and n-links with the minimal cost so that all terminals are separated and each labeling node $L_i(m,n)$ only connects to one terminal through a t-link.

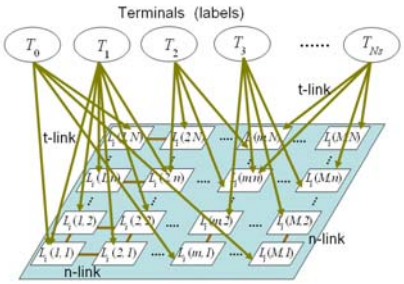


Figure 8. The graph cut model for the optimal labeling.

Moreover, in the graph cuts algorithm, the initial guess of L is obtained by finding the labeling image of each camera view that maximizes the probability function in (19) under the status hypothesis S^H. That is, we find the initial labeling image $L_i^{ini}(m,n)$ of the ith camera view such that

$$L_i^{ini}(m,n) = \arg\max_{L_i} \prod_m \prod_n p(L_i(m,n) \mid S = S^H). \quad (24)$$

Among all status hypotheses, the status hypothesis that achieves the maximum posterior probability is picked as the optimal status combination S^*. The optimal labeling of S^* is then inferred as the optimal labeling L^*.

B.4. 3-D Target Model Refinement

In our system, the 3-D model of each target is a pillar model with parameters height (H) and width (R) standing at a location X on the ground plane. However, different targets may have different heights and widths. In our system, we treat these model parameters as latent random variables and introduce an EM-based algorithm to iteratively refine the parameters.

Initially, the proposed EM algorithm adopts the pre-trained probability distributions $p(H)$ and $p(R)$ to model the uncertainty of each target's height and width. Since the BHF framework

combines not only the 3-D scene priors and target priors but also the observed image data and the corresponding foreground detection result, the optimal target labeling reveals personal properties of each detected target. Hence, based on the labeling results, we update the probability distributions of H and R and establish personalized 3-D models gradually.

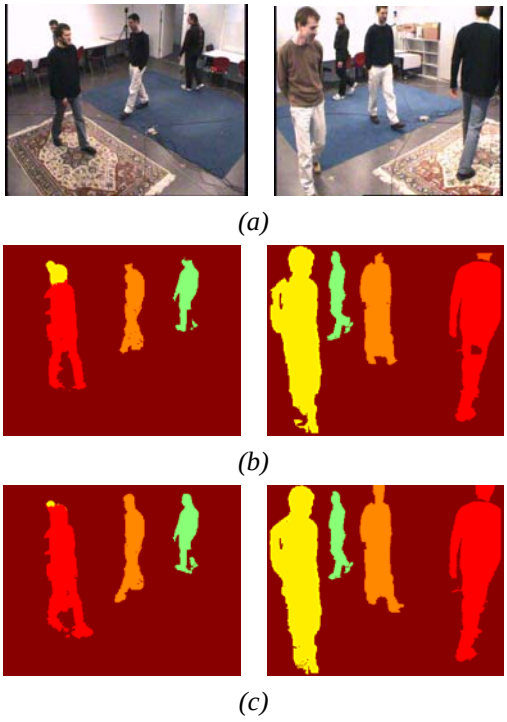


Figure 9. (a) Two camera views. (b) Labeling results without target model refinement. (c) Labeling results with target model refinement.

In detail, we assume the model height and width are independent and can be refined separately. With Bayes rule, the refinement of the posterior probability is defined as

$$p(Q_k^r \mid L^r) \equiv C \cdot p(L^r \mid Q_k^r) \cdot p(Q_k^r) \qquad (25)$$

where

$$p(Q_k^r) = \begin{cases} p(H) \text{ or } p(R) & \text{if } r = 1 \\ p(Q_k^{r-1} \mid L^{r-1}) & \text{otherwise} \end{cases}$$

In (25), L^r denotes the labeling results of multiple image views at the rth iteration. The notation $Q_k^r \in \{H_k^r, R_k^r\}$ represents the height or the width of the kth target at the rth iteration. Also, C is a normalization constant, $p(L^r \mid Q_k^r)$ is the likelihood term to be defined later, and $p(Q_k^r)$ is the prior probability of Q_k^r. In our system, we treat $p(Q_k^{r-1} \mid L^{r-1})$ as the prior information propagated from the previous iteration. For the first iteration, $p(Q_k^1)$ is set to the pre-trained probability.

To formulate the likelihood term $p(L^r \mid Q_k^r)$, we project the pillar model at the ground position of the kth target with height H_k^r and width R_k^r onto multiple camera views to check the overlapping regions with the labeling results. Ideally, if a more accurate model parameter is chosen, the projected region will better fit the labeling result. Hence, the likelihood is termed as

$$p(L^r \mid Q_k^r) = \left(\prod_i \prod_{m,n \in A_i^k} \left(p_{m,n}^i(l) \right) \right)^{1/N} \qquad (26)$$

In (26), A_i^k is the projected region of the kth target onto the ith camera view. $p_{m,n}^i(l)$ is the probability of labeling the pixel at (m,n) with the ID l. N is the total number of pixels within the projected region. Since different $\boldsymbol{Q}_k^r$ may generate different projected regions, we take the $1/N$ power for normalization. Moreover, we assume the statuses of different labeling pixels are independent and we evaluate only those pixels inside the projected region of the kth target. In principle, the label ID tends to be T_k. Hence, $p_{m,n}^i(l)$ has a higher probability if l equals to T_k and has a lower probability if l equals to T_0. Occasionally, owing to occlusion, l may equal to some other foreground target. In this case, we assign $p_{m,n}^i(l)$ to be an intermediate value. That is, we define $p_{m,n}^i(l)$ as

$$p_{m,n}^i(l) = \begin{cases} \lambda \cdot e^x & \text{if } l = T_k \\ \lambda \cdot e^y & \text{if } l = T_0 \\ \lambda \cdot e^z & \text{otherwise} \end{cases} \qquad (27)$$

where λ is a normalization constant to make the probability summation equal to 1. In (27), *x*, *y*, and *z* are the weighting parameters with the value 5, -3, and 0, respectively, which satisfies the relation $x > z > y$. If we rewrite (24) based on (25), we get a likelihood form as follows

$$p(L^r \mid Q_k^r) = \lambda \cdot \exp\{\frac{1}{N}(x \cdot N_k + y \cdot N_0 + z \cdot N_{other})\} \quad (28)$$

where N_k, is the number of T_k-labeled pixels, N_0 is the number of T_0-labeled pixels, and N_{other} is the number of the other pixels inside the projected regions in all camera views. Basically, (28) measures the degree of matching by accumulating the weighted sum of different labeling pixels inside the projected regions with the weighting parameters (*x*,*y*,*z*). Once the likelihood term $p(L^r \mid Q_k^r)$ is determined, the refined probability distribution of the height and width of the *k*th target height in the current iteration can be obtained based on (25). The refined models $p(H_k^r \mid L^r)$ and $p(R_k^r \mid L^r)$ are inputted to the proposed BHF for the next iteration of the optimal object labeling. In our experiments, it usually takes only 2 to 3 iterations to construct the refined target model.

4. MULTI-VIEW FACE DETECTION FRAMEWORK

After the position estimation, we identify the ground locations *X* of detected candidate targets on the 3-D ground. However, the head location of each candidate is still unknown. Even so, the extracted locations are useful for speeding up head finding. In next subsections, we aim to search head positions and determine the face directions. We will formulate the problem and explain our multi-view face detection framework.

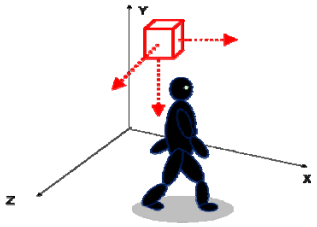


Figure 10. The sliding cube in 3-D space.

A. Finding Target Heads and Face Directions

Many detection algorithms are already proposed for multi-view face detection. Most of these methods based on some learning approaches to train suitable detectors. After the training process, the trained detectors are able to detect specific object based on the sliding window approach in the image. However, because of some reasons, this sliding window approach in 2-D image may not be suitable for our application. First, it would not be easy to train a high accuracy detector for all face views. Second, from time to time, we need to search all scales to detect faces with different sizes including very small faces in the image. These small faces are usually too small to correctly identify. Also, the heavy searching time is unwelcome. Third, there could be some occlusions in the scene. Sometimes, we may need to detect faces that are incompletely observed in the image view. Due to the above reasons, detecting face directly in 2-D images usually generates many inevitable false detection and false rejection.

A multi-camera system may provide us more information about the scene and could theoretically decrease the false positive rate and increase the detection rate. However, the performance of the multi-camera system depends heavily on the way we utilize the 3-D geometric information. A conventional way is to detect faces in each 2-D image and then the 2-D detection

results are projected back to the 3-D space for the final decision. Strictly speaking, this intuitive way is too ideal to be used in practical applications. This approach could work only when the detection rate of the 2-D face detector is high enough and the false alarm is low. Otherwise, the presence of plentiful false positives and false negatives would make the inference in the 3-D space very complicated and mistake-prone. The wrong information comes from 2-D images would accumulate and generate a lot of ambiguous results in the 3-D space.

Instead of searching and detecting the targets in 2-D images and then combining the outcome of each camera view in the 3-D space, in our system, we try to search and detect the targets in the 3-D space. This approach is like an extension from the 2-D sliding window approach to a 3-D sliding cube approach. In Figure 10, we illustrate this concept.

We now slide a cube in the 3-D space and determine 3-D head locations and face directions. However, we do not have the reconstructed 3-D scene for the 3-D based detection. In practical situations, what we have are the observations of 2-D images. Also, the reconstruction of the 3-D scene from all 2-D images is not reliable due to the limited number of cameras and the insufficient information from 2-D images. Hence, in our approach, we directly look for supports in the 2-D images. Here, we well utilize the geometric connection between 3-D space and 2D images according to the process of camera calibration beforehand. Based on the prior knowledge of 3-D geometric space, we generate hypotheses of head locations and face directions in the 3-D domain and base on the observed data from multiple 2-D images to make the final decisions. In each hypothesis, we assume the target face is at a specific location and direction in the 3-D space and confirm this hypothesis in its corresponding 2-D image regions. Figure 11 shows an example of this process. In this example, a person is assumed to walk in a 3-D surveillance zone and a cube is sliding in the 3-D space to find where the head and what face direction of this person are. If the cube is slid to a suitable location that contains the human's head, the corresponding regions in 2-D images will fit the face portion with a proper face view. On the contrary, if the cube is at a place without any person's head, then the corresponding region will map to the background region and will not fit the face portion in each camera view.

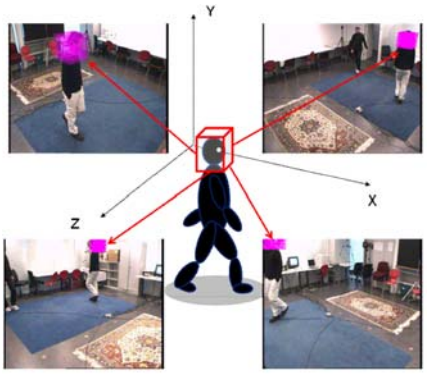


Figure 11. A 3-D sliding cube approach for the finding of human heads and face directions.

B. Problem Formulation

Based on all the discussion in the previous sections, we now formulate our system goal as an optimization problem in a unified manner. Until now, we have detected N_T candidate targets and their ground locations $\{X_i\}_{i=1\sim N_T}$. Here, we use T_i to represent the ID of the ith target. For each target, we still need to determine its head location and face orientation. As mentioned before, we use a sliding cube approach to find the optimal location (l*) and orientation (h^*) of the target T_i. Here, we define the optimal solution as

$$(h_{T_i}^*, l_{T_i}^*) = \arg \max_{h \in H \;\; l \in L|_{T_i}} p(h, l \mid D, I, X_i, C). \quad (29)$$

To numerically analyze this optimization problem in (29), we uniformly quantize the solution spaces of 3-D head positions and face orientations and denote the spaces as L and H respectively. In our system, the interesting 3-D space L, bounded by the surveillance zone and a user-defined height 200cm, is divided into 100x100x50 cubes. The orientation space H, ranging from zero to 360 degree, is divided into eight face directions. We also define other notations in (29) as below:

(D): The set of eight image-based face classifiers pre-trained for different face orientations.

(I): The set of multi-camera image views.

(X_i): The ground location of candidate target T_i.

(C): Camera layout and geometry information.

($L|_{T_i}$): The possible 3-D head positions of candidate target T_i.

Here, $L|_{T_i}$ needs to be detailed. In our system, X_{N_T} indicates the set of estimated ground positions of the N_T detected targets in the 3-D space and could be utilized to reduce the solution space of the head locations. We Assume $X_i = \{x_i, y_i, 0\}$, where (x_i, y_i) is the ground position of the ith detected target. If we know the mean of human height is z_0 beforehand, we can reduce the interesting 3-D head positions of the target T_i, and define the reduced space $L|_{T_i}$ as

$$L|_{T_i} = \left\{ \begin{array}{r} x_i - s/2 \le x \le x_i + s/2 \\ (x, y, z) \,\middle|\, y_i - s/2 \le y \le y_i + s/2 \\ z_0 - s/2 \le z \le z_0 + s/2 \end{array} \right\}. \quad (30)$$

In (30), s defines a search range and is determined by the average size of a 3-D head. In our system, we set s *as* three times of the average size in order to account for the uncertainty. Also in (8), the average 3-D human height z_0 is obtained through statistical training. In Figure 12, we illustrate the reduced position space given the plane location (x_0, y_0, 0) of the first detected target.

To solve (29), we still need to define the calculation of $p(h,l|D,I,X_i,C)$. In detail, for each hypothesis (h,l) in the 3-D space, we project a cube at 3-D location l onto 2-D images to locate the focused patches and also generate the expected face orientations in different camera views based on h, I, and C. Here, we use Equation (31) to expresses the 3-D to 2-D projection process, where the function $B(.)$ projects the 3-D cube and generates the expected face direction; $I_{n,ED,l}$ indicates the projected image patch with the expected face direction ED_n in the nth camera views of our four-camera system

$$I_{n,ED,l} = B(l, h \mid I, C) \;\; \forall l \in L|_{T_i} \quad n = 1, 2, 3, 4 \quad (31)$$

For each camera view, we based on the expected face direction ED to select the corresponding face classifier from the classifier set D. By feeding the image patch $I_{n,ED,l}$ into the selected classifier, we could evaluate the likelihood $p_{n,l,h}$ of the hypothesis (h,l) based on information form this camera view. This process could be defined as

$$p_{n,l,h} = D(I_{n,ED,l}; ED_n) \quad n = 1, 2, 3, 4. \quad (32)$$

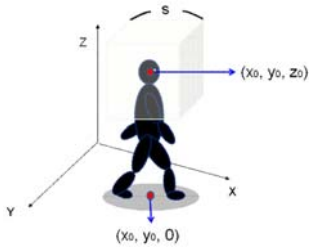


Figure 12. The reduced position space given a detected target location (x_0, y_0, 0).

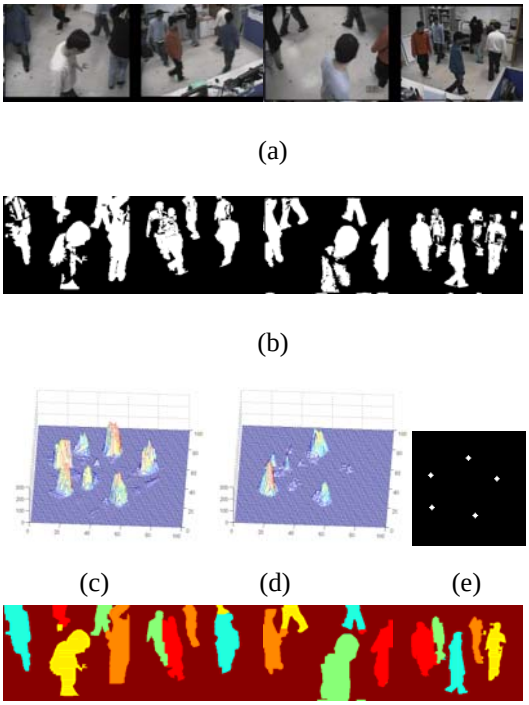
Note that ED_n determines one of the eight pre-trained face classifiers from the classifier set D. By combining the likelihoods from all camera views, we then define $p(h,l|D,I,X_i,C)$ as

$$p(h,l \mid D,I,X_i,C) = \prod_{n=1}^{4} p_{n,l,h} \tag{33}$$

Finally, we exhaustively search the solution spaces H and $L|_{T_i}$ in order to determine the optimal head location (l^*) and face orientation (h^*) for target T_i in (29). Thanks for the pre-process of 3-D position estimation step introduced in Sec. 3, the solution space $L|_{T_i}$ is greatly reduced and the searching process is speeded up. Moreover, based on the automatically extracted target number N_T, we know the number of targets we need to search. Unlike many conventional face detection methods, no more lots of detected windows around a face region but only N_T face window with suitable scales are detected. In the next subsection, we would like to introduce how we train the classifier set D.

5. EXPERIMENTS RESULTS

To test our system over real video sequences, we set up four static cameras in our lab to capture test sequences. In our test sequences, the coverage is about 4.5m by 4.5m, with 3 to 5 targets moving within the zone. A set of snap shots with 5 persons inside the scene are shown in Figure 13(a). On the other hand, we also tested our system over the video sequences provided by the M2Tracker project [11] and the sequence used in Fleuret's papers [12-14]. Both video sequences are publicly available. The M2Tracker sequence was captured by 15 synchronized cameras over a 3.0m by 3.0m area, while Fleuret's sequence was captured by four synchronized cameras in a 12.8m^2 room.



(a) (b) (c) (d) (e) (f)

Figure 13. One experiment result of our LAB sequence. (a) Four camera views. (b) Foreground detection images. (c) TDP distribution. (d) Voxel histogram based on visual-hull reconstruction. (e) Bird-eye view of target location. (f) Labeling and correspondence of targets in pseudo-color.

For each sequence, the cameras have been geometrically calibrated with respect to a world coordinate system. Except the M2Tracker sequence, each video sequence contains more than 300 frames. Especially, Fleuret's video sequence contains as many as 3900 frames. For the evaluation of object ground location, we acquired the ground truth of M2Tracker sequence from Dr. Guan Li, the author of [15]. To establish the ground truth of Fleuret's sequence, we manually identified the image positions of human necks and used them as the corresponding points among images. By backprojecting these corresponding points onto the 3-D space, object locations on the ground plane can be estimated. Here, we manually created a ground truth frame

for every other 25 frames. To see the details of our experimental results, please visit our website [16].

To understand the process of head localization and multi-view face detection in our system, we show the projected windows under different 3-D location hypotheses in the four camera views. In Figure. 14(a), owing to a correct 3-D location hypothesis, the projected windows match the face regions well, while in Figure 14(b), the projected windows shift away from the correct face regions due to the wrong 3-D location hypothesis. To quantitative compare, we draw the calculated likelihood values under both the correct 3-D location and the wrong 3-D location over eight hypotheses of face orientations as shown in Figure. 14(c). Please note the blue curve, indicating the values under the correct location, is always higher than the green curve, representing the values under the wrong location. Also, the largest likelihood value over the blue curve indicates the optimal hypothesis of face orientation.

We tested our system over the video sequence provided by Fleuret's work [17]. Note the sequence contains more than 2000 frames. To quantitatively evaluate the detection and correspondence performance, false positive rate (FPR) and false negative rate (FNR) are used. In our system, the target detection and correspondence are defined as "correct" when the projected regions of the detected target in all camera views intersect the same individual and the detected face directions match the ground truth. Based on this definition, the calculated FPR and FNR of all tested sequence are 0.065 and 0.023.

We also show some detection results in Figure. 15. To clearly present our outputs, we use bounding boxes with different colors to indicate different targets. We also mark the detected face direction onto the bird-eye view of the surveillance zone. In this example, there are two persons in the scene. As shown in the figure, our system can detect faces and identify the face directions even if some serious occlusion occurs or someone is out of image view. In Figure. 15(a), there is an occlusion case in the top-left image and there is a missing person in the lower-right image. For this example, our system can still find the approximate locations of the faces and the face directions, as shown in Figure. 15(b). Another experimental result is illustrated in Figure. 15(c-d).

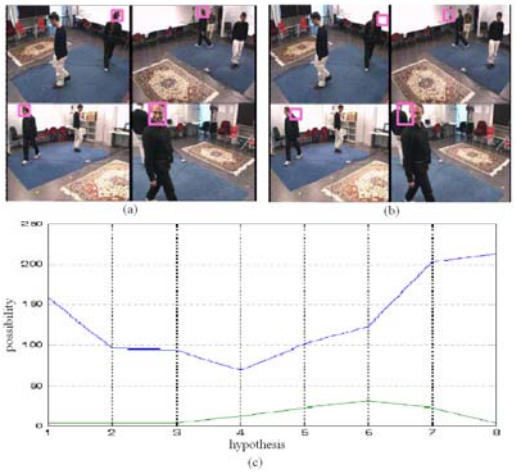


Figure 14. (a) Detection result at a correct 3-D position. (b) Detection result at an incorrect 3-D position. (c) The blue line corresponds to the likelihood values of eight hypotheses of face orientations at the correct position, and the green line corresponds to he likelihood values of eight hypotheses at the incorrect position.

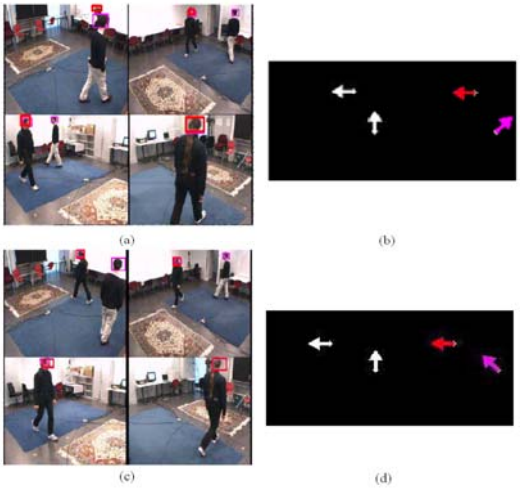


Figure 15 (a) Multi-view face detection results with inter-object occlusion. (b) The bird-eye view of detected face directions of (a). (c) Another Multi-view face detection results. (d) Detected face directions of (c). Note white arrows indicate the ground truth. Colored arrows indicate our detection results.

6. REFERENCES

[1] Ching-Chun Huang, and Sheng-Jyh Wang, "A Monte Carlo Based Framework for Multi-Target Detection and Tracking Over Multi-Camera Surveillance System," *European Conference on Computer Vision Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications*, October 12-18, 2008.

[2] Ching-Chun Huang, and Sheng-Jyh Wang, "Moving Targets Labeling and Correspondence over Multi-Camera Surveillance System Based on Markov Network," *IEEE International Conference on Multimedia and Expo*, June 28-July 3, 2009.

[3] Q. Yu, G. Medioni, I. Cohen, "Multiple Target Tracking Using Spatio -Temporal Markov Chain Monte Carlo Data Association," *IEEE International Conference on Computer Vision and Pattern Recognition*, 2007.

[4] B. Georgescu, I. Shimshoni, P. Meer, "Mean Shift Based Clustering in High Dimensions: a Texture Classification Example," *IEEE International Conference on Computer Vision*, 2003.

[5] T. Boykov, O. Veksler, R. Zabih, "Markov Random Fields with Efficient Approximations," *IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 648-655, June 1998.

[6] Pedro F. Felzenszwalb and Daniel P. Huttenlocher, "Efficient Belief Propagation for Early Vision," *International Journal of Computer Vision*, Vol. 70, No. 1, October 2006.

[7] R. S. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother, "A Comparative Study of Energy Minimization Methods for Markov Random Fields with Smoothness-Based Priors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume 30, Number 6, June 2008 pp 1068-1080.

[8] Y. Boykov, O. Veksler and R. Zabih, "Efficient Approximate Energy Minimization via Graph Cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.3, pp.1222-1239, 2001.

[9] Vladimir Kolmogorov and Ramin Zabih, "What Energy Functions can be Minimized via Graph Cuts?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, pp. 147-159, February 2004.

[10] Yuri Boykov and Vladimir Kolmogorov, "An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, September 2004.

[11] A. Mittal and L. Davis, "M2Tracker: A Multi-View Approach to Segmenting and Tracking People in a Cluttered Scene", *International Journal of Computer Vision*, Vol. 51, Issue 3, pp. 189-203. Feb. 2003.

[12] F. Fleuret, R. Lengagne, and P. Fua, "Fixed Point Probability Field for Complex Occlusion Handling," *IEEE International Conference on Computer Vision*, 2005.

[13] J. Berclaz, F. Fleuret, and P. Fua, "Robust People Tracking with Global Trajectory Optimization," *IEEE International Conference on Computer Vision and Pattern Recognition*, 2006.

[14] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multi-Camera People Tracking with a Probabilistic Occupancy Map," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 30, Issue. 2, pp. 267 - 282, February 2008.

[15] Li Guan, Jean-Sebastien Franco, and Marc Pollefeys, "Multi-Object Shape Estimation and Tracking from Silhouette Cues", *IEEE International Conference on Computer Vision and Pattern Recognition*, Anchorage, Jun. 2008.

[16] Ching-Chun Huang. (2010). *Huang's Projects* [Online]. Available at http://140.113.238.220/~chingchun/projects.html.

[17] A. Mittal and L. Davis, "Unified Multi-camera Detection and Tracking Using Region-matching," *in Proceedings of IEEE Workshop on Multi-Object Tracking*, pp. 3–10, Vancouver, BC, Canada, July 2001.

計畫成果自評

在本計畫中，我們驗證了以貝氏階層式結構為基礎的影像分析架構可以有效地應用到視訊監控的分析與應用上。透過此架構，我們將像素層級的色彩資訊、像素間的區域層級資訊、以及以物體為基本單位的物件層級資訊有系統地整合在一起，這樣的整合讓系統可以擁有更多的資訊，並可以針對較複雜的影像內容進行準確的推論分析。

在多攝影機視訊監控系統中，我們自動地定位、標記、與對應在不同攝影機監控範圍內的多個物體，同時有效壓抑因為幾何深度上的不確定性所產生的假物體。多攝影機視訊監控系統在真實的應用場景中，往往面臨一些具挑戰性的議題: (a) 場景中未知物體的數量; (b)物體間的相互遮蔽; 以及(c)假物體的出現。有別於過去的方法，我們提出了一套包含資訊整合與場景推論的兩步驟策略。在資訊整合的步驟中，我們整合來自多攝影機的資訊以建立一機率分佈，藉以描述物體出現於地面某一位置的可能性。在場景推論的步驟中，我們應用貝氏階層式結構將場景模型納入考量，透過此結構，我們將物件在影像內的標記議題、物件在多攝影機間的對應議題、以及假物件的消除議題整合為單一的最佳化問題。此外，我們進一步採用期望-最大化架構來調整出更好的物體三維模型，透過貝氏階層式結構與期望-最大化架構的結合，我們可以得到更好的系統效能。實驗結果顯示，我們的系統可以自動地決定場景中的運動物體數量、有效地標記並對應出不同攝影機影像中的多個物體、準確地定位物體在三維場景中的位置、並且能有效地清除假物件。

同時我們以此貝氏階層式結構為基礎。提出一套應用於多台攝影機之多角度人臉偵測系統。此系統可根據多攝影機擷取的影像偵測出影像中的人臉位置，並得到在三維空間中人臉方向的鳥瞰圖。實驗結果顯示，我們的系統在物體相互遮蔽，以及前景區域與背景區域因為外貌相似而混淆的情況下，仍然可以得到好的效果。

此為三年期計畫，若包含前兩年之執行成果，本計畫在三年期間一共完成以下數篇學術論文。

1. 國際期刊論文2篇

- Hsien Chen and Sheng-Jyh Wang, "An Efficient Approach for Dynamic Calibration of Multiple Cameras," *IEEE Transactions on Automation Science and Engineering*, VOL. 6, NO. 1, pp. 187-194, Jan. 2009.
- Ching-Chun Huang and Sheng-Jyh Wang, “A Bayeisan Hierarchical Framework for Multitarget Labeling and Correspondence with Ghost Suppression over Multicamera Surveillance System”, to appear on *IEEE Transactions on Automation Science and Engineering.*

2. 國際研討會論文5篇

- Ching-Chun Huang and Sheng-Jyh Wang, “A Monte Carlo Based Framework for Multi-Target Detection and Tracking Over Multi-Camera Surveillance System,” *in Proceedings of the 10th European Conference on Computer Vision (ECCV) Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications, Marseille*, France, Oct. 2008.

- Ching-Chun Huang and Sheng-Jyh Wang, “Moving Targets Labeling and Correspondence over Multi-Camera Surveillance System Based on Markov Network”, *in Proceeding of 2009 IEEE International Conference on Multimedia and Expo*, June 2009.
- Ching-Chun Huang, Wei-Chen Chiu, Sheng-Jyh Wang, and Jen-Hui Chuang, “Probabilistic Modeling of Dynamic Traffic Flow across Non-overlapping Camera Views”, *in Proceedings of International Conference on Pattern Recognition*, Istanbul, 2010.
- Ching-Chun Huang and Sheng-Jyh Wang, “A Cascaded Hierarchical Framework for Moving Object Detection and Tracking”, *in Proceedings of IEEE International Conference on Image Processing*, Sep., 2010.
- Ching-chun Huang, Jay Chou, Jia-hau Shiu, and Sheng-Jyh Wang, “Multi-view Face Detection Based on Position Estimation over Multi-camera Surveillance System”, *to appear in Proceedings of SPIE*, 2012.

3. 國內研討會論文2篇

- Ching-Chun Hsiao and Sheng-Jyh Wang, “Model-Based pose estimation for multi-camera motion capture system,” *in Proc. Computer Vision, Graphics, and Image Processing*, Taiwan, 2008.
- Po-Kai Fan and Sheng-Jyh Wang, “Coordination of PTZ cameras Based On Particle Swarm Optimization for cooperative video surveillance,” *in Proc. Computer Vision, Graphics, and Image Processing*, Taiwan, 2008.