Video-based Face Recognition

Rama Chellappa (UMD) and
Pavan Turaga (ASU)

Why video-based face recognition?

- Because it's there (Willie Sutton, a famous back robber)
- Video-based face recognition can incorporate appearance and motion cues
- More data
- VFR can work under adverse acquisition conditions (Alice O'Toole)
- Still face recognition is at a critical point
  - Close to 100% performance with standard data sets (we are becoming like the iris folks!)
  - Performance on uncontrolled faces not good enough.

Video-based recognition: challenges

Quality of video is low

- Under non-ideal acquisition condition
- Objects are not co-operative
- Pose and illumination variations dominate

Face images are too small

- Not suitable for many methods, e.g., local feature methods

Characteristics of faces

- Relatively easy for detection, but hard for recognition

Outline

- Appearance-based simultaneous tracking and recognition
- Video dictionaries for face recognition
- Detect and associate faces in a video
- Face tracking and recognition in a camera network
- Albdeo-based simultaneous tracking and recognition
- Manifold-based VFR
  - Empirical manifolds
  - Analytic manifolds
  - Appearance, shape and landmarks-based manifolds
  - VFR using manifolds

Simultaneous tracking and recognition

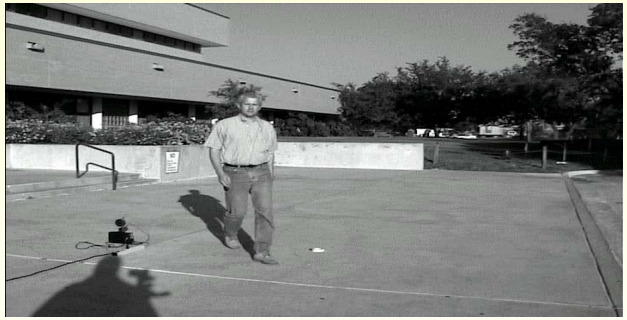
- Tracking and recognition of moving objects from moving cameras is a challenging problem.
- Pose, scale and illumination variations.
- More than one frame is available.
- Tracking using online appearance modeling.
- Simultaneous tracking and recognition.

Video-based face recognition

Gallery

$\{ I_1, I_2, \ldots, I_N \}$

Probe



Faces in probe videos

S. Zhou, R. Chellappa, and B. Moghaddam, "Visual tracking and recognition using appearance-adaptive models in particle filters," *IEEE Trans. on Image Processing* , vol. 11, pp. 1434-1456, Nov. 2004

Tracking-then-recognition v.s. tracking-and-recognition approaches

Tracking-then-recognition	Tracking-and-recognition
Utilize temporal information for tracking only	Utilize temporal information for tracking and recognition
Wait for good frames	Recursively process each frame
Register the frames	Registration is automatic
Essentially still-image-based face recognition	Simultaneous tracking and recognition

Terminology

- Gallery set: $\{ I_1, I_2, \ldots, I_N \}$.
- Observations: $\{y_1, y_2, \ldots, y_T\}$.
- Transformation parameters: $\{q_1, q_2, \ldots, q_T\}$.
- Identity variable: $n \in \{1,\ldots,N\}$.

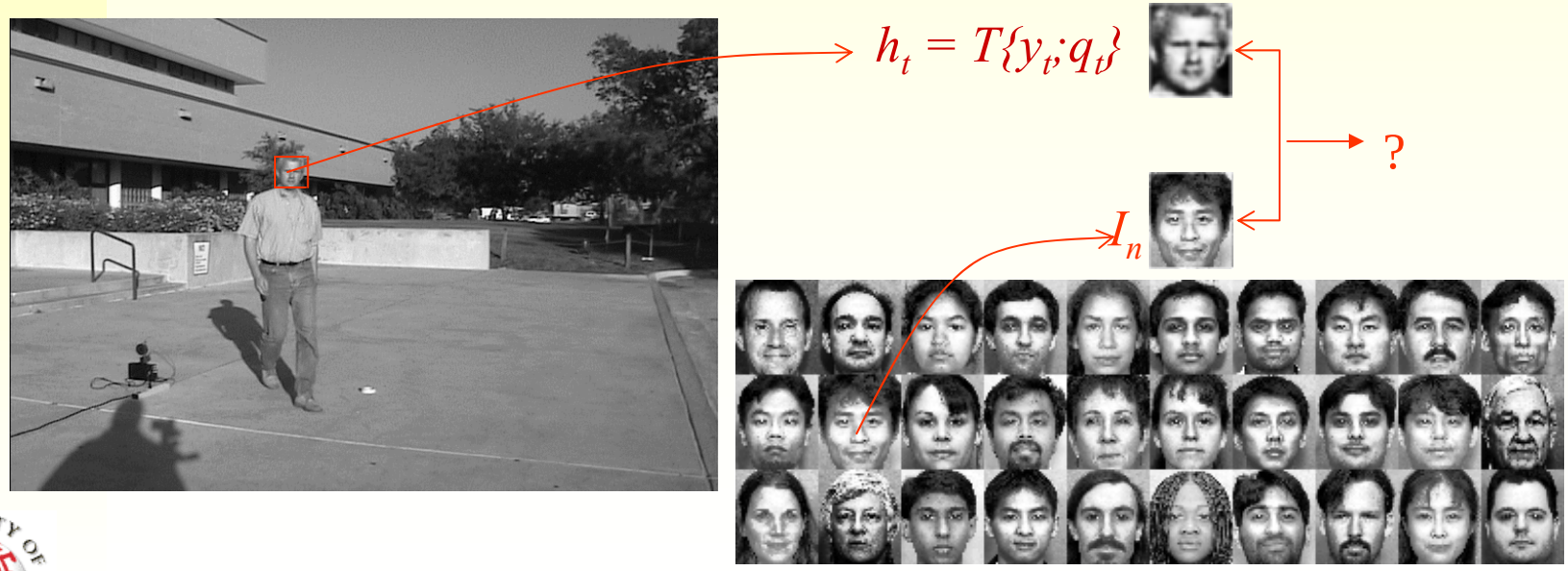


Image y_t

Time series state space model for tracking-and-recognition

Motion equation:

$q_t = g(q_{t-1}) + u_t$.

Identity equation:

$n_t = n_{t-1}$.

Observation equation:

$h_t \equiv T\{y_t; q_t\} = I_{nt} + v_t$

- Prior distribution:
- $p(q_0)$, $p(n_0)$.
- Noise distribution:$p(u_t)$:
- State transition prob. $p(q_t|q_{t-1})$
- likelihood $p(y_t|q_t,n_t)$.
- Statistical independence

Solving the model

- Computing the posterior $p(n_t, q_t | y_{1:t})$.
  - Marginal posterior probability.
  - $p(n_t | y_{1:t})$: for recognition.
  - $p(q_t | y_{1:t})$: for tracking.
- Nonlinear, Non-Gaussian.
  - No analytic solution.
- Sequential Importance Sampling (SIS).
- Efficient computation.

Stochastic appearance tracking

- Appearance tracking is a stochastic process for modeling inter-frame motion and appearance changes
  - Video frame $\{ Y_1, Y_2, \ldots, Y_t, \ldots \}$
  - Motion parameter $\{ q_1, q_2, \ldots, q_t, \ldots \}$
  - State equation (motion model):

$$q_t = F_t(q_{t-1}, U_t)$$

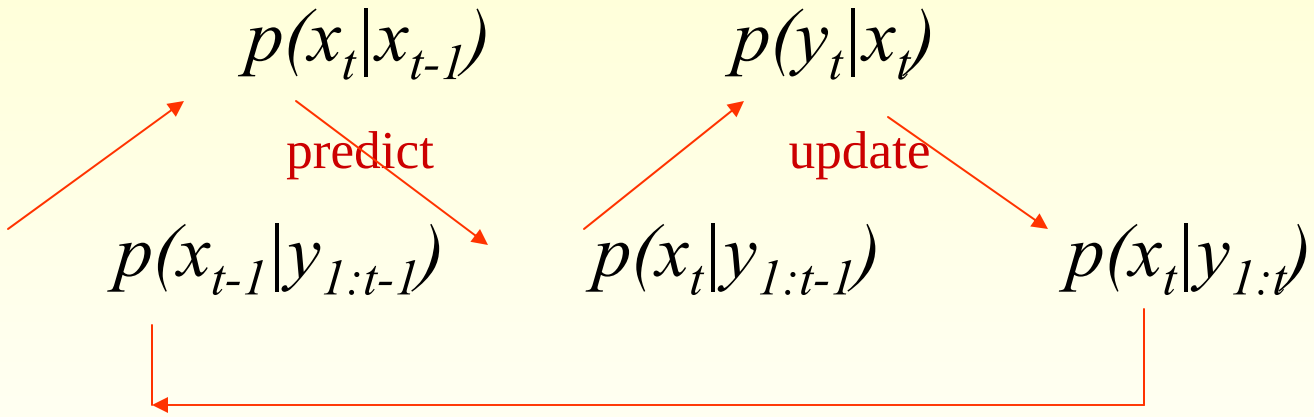
  - Observation equation (model): $Y_t = G_t(q_t, V_t)$

Statistical Inference & particle filtering

- Statistical inference
  - Computing posterior probability $p(q_t|Y_{1:t})$
- Particle filtering (PF)
  - PF approximates $p(q_t|Y_{1:t})$ using a set of weighted particles $\{q_t^{(j)}, w_t^{(j)}; j=1,...,J\}$
  - Two steps: (i) propagate the particles governed by the motion model; (ii) update the weights using the observation model.
  - The state estimate q_t^* can be a MMSE, MAP, or other estimate based on $p(q_t|Y_{1:t})$ or $\{q_t^{(j)}, w_t^{(j)}; j=1,...,J\}$.

Sequential importance sampling

- SIS (or particle filter) is a ‘sample’ version of Kalman filter.



- Kalman filter: mean and covariance.
- SIS (particle filter): samples and weights.

Sequential importance sampling

$$\{\chi_t^{(j)}, w_t^{(j)}\}_{j=1}^m \sim \pi_t(\chi_t) \xrightarrow{SIS} \{\chi_{t+1}^{(j)}, w_{t+1}^{(j)}\}_{j=1}^m \sim \pi_{t+1}(\chi_{t+1})$$

SIS steps: for $j = 1, \cdots, m$,

(A) Draw $X_{t+1} = \mathbf{x}_{t+1}^{(j)}$ from $g_{t+1}(\mathbf{x}_{t+1}|\mathcal{X}_t^{(j)})$. Attach $\mathbf{x}_{t+1}^{(j)}$ to form $\mathcal{X}_{t+1}^{(j)} = (\mathcal{X}_t^{(j)}, \mathbf{x}_{t+1}^{(j)})$.

(B) Compute the "incremental weight" u_{t+1} by

$$u_{t+1}^{(j)} = \frac{\pi_{t+1}(\mathcal{X}_{t+1}^{(j)})}{\pi_t(\mathcal{X}_t^{(j)}) g_{t+1}(\mathbf{x}_{t+1}|\mathcal{X}_t^{(j)})}$$

and let $w_{t+1}^{(j)} = u_{t+1}^{(j)} w_t^{(j)}$.

$$g_{t+1}(x_{t+1} \mid \chi_t) = q_t(x_{t+1} \mid x_t) \Rightarrow u_{t+1} \propto p(y_{t+1} \mid x_{t+1}, Y_t) = p(y_{t+1} \mid x_{t+1})$$

Observation equation in regular visual tracking

- Observation equation
  - $T\{Y_t; q_t\} \equiv Z_t = A_t + V_t$. A_t : appearance model
- Practical appearance models
  - Fixed appearance model; $A_t = A_0$
    - Hard to handle the appearance changes in the video though stable
  - 'Most-recent' appearance model; $A_t = T\{Y_t; q_t^*\} \equiv Z_t^*$
    - Susceptible to drift though able to follow appearance changes
  - A good appearance model should be a compromise

Adaptive visual tracking

- Strategy: appearance-adaptive
  - State and observation models adaptive to the appearances in the video
- Adaptive observation model
  - $T\{Y_t; q_t\} \equiv Z_t = A_t + V_t$
  - A_t is a mixture appearance model (MAM) adaptive to all past observations
- Adaptive motion model
  - Time-varying Markov model: $q_t = q_{t-1} + U_t$
  - Adaptive noise variance; $U_t = n_t + r_t U_0; U_0 \sim N(0, S_0)$
  - The mean n_t and the 'variance' function r_t, both time-varying, adapt to the incoming frame Y_t

Mixture appearance model (MAM) - 1

- Mixture of 3 components: stable, wandering, fixed
  - Stable (‘*S*’) component captures a slowly-varying structure in the appearance.
  - Wandering (‘*W*’) component captures a rapidly-varying structure in the appearance.
  - Fixed (‘*F*’) component, which is optional, captures a constant structure in the appearance.
  - Each component has *d* pixels, assumed to be independent.

MAM -2

- A_t : $\{m_{i,t}(j),\ s^2_{i,t}(j),\ m_{i,t}(j);\ i=w,s,f;\ j=1,\ldots,d\}$
  - Mixture centers: $m_{i,t}(j)$
  - Variance: $s^2_{i,t}(j)$
  - Mixing probabilities: $m_{i,t}(j)$
  - Likelihood function: mixture of Gaussians
    - $p(Y_t|q_t) = p(Z_t|q_t)$
      $= P_{j=1:d}\{S_{i=w,s,f} m_{i,t}(j)\ N[Z_t(j);m_{i,t}(j),s^2_{i,t}(j)]\}$
    - Normal $N[x;m,s^2] = (2ps^2)^{-1/2}exp(-r((x-m)/s));$ $r(y)=y^2/2$

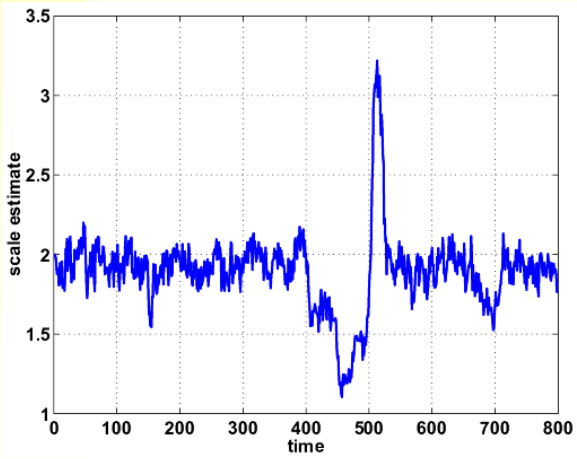
MAM update

- Update $A_t \rightarrow A_{t+1}$ using the tracked patch $Z_t^* \equiv T\{Y_t; q_t^*\}$
  - Stable component is 'exponentially' updated with rate a
  - Wandering component is completely refreshed
  - Fixed component is unchanged

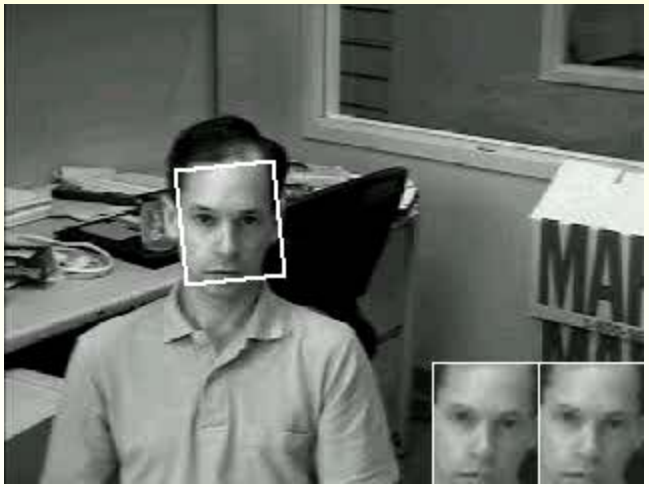
MAM update - 2

- Update equations
  - Posterior probabilities: $o_{i,t}(j) \propto m_{i,t}(j)\ N[Z_t^*(j); m_{i,t}(j), s^2_{i,t}(j)]$
  - 1st and 2nd moment: $M_{p,t+1} = aZ_t^*(j)o_{s,t}(j) + (1-a)M_{p,t}(j);\ p=1,2$
  - Mixing probabilities: $m_{i,t+1}(j) = a\ o_{i,t}(j) + (1-a)\ m_{i,t}(j)$
  - Mixture centers and variances
    - $m_{s,t+1}(j) = M_{1,t+1}/m_{s,t+1}(j);\ s^2_{s,t+1}(j) = M_{2,t+1}/m_{s,t+1}(j) - m^2_{s,t+1}(j)$;
    - $m_{w,t+1}(j) = Z_t^*(j);\ s^2_{w,t+1}(j) = s^2_{w,1}(j);$
    - $m_{f,t+1}(j) = F_1(j);\ s^2_{f,t+1}(j) = s^2_{f,1}(j);$

Face tracking

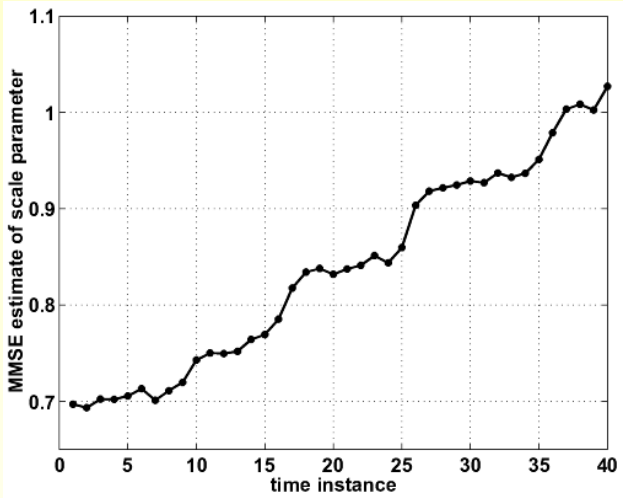


Scale estimate over time

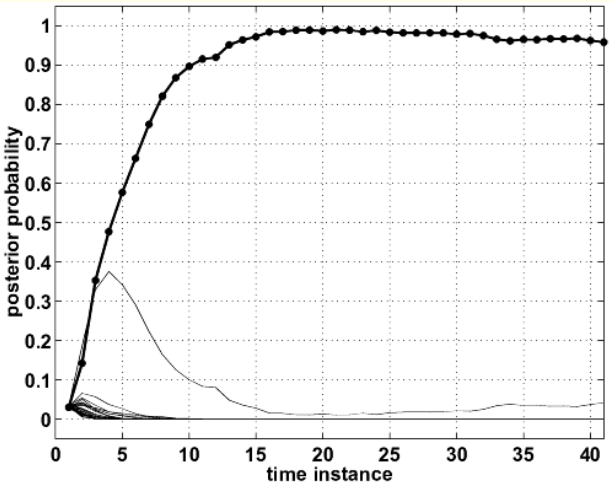


Without occlusion analysis

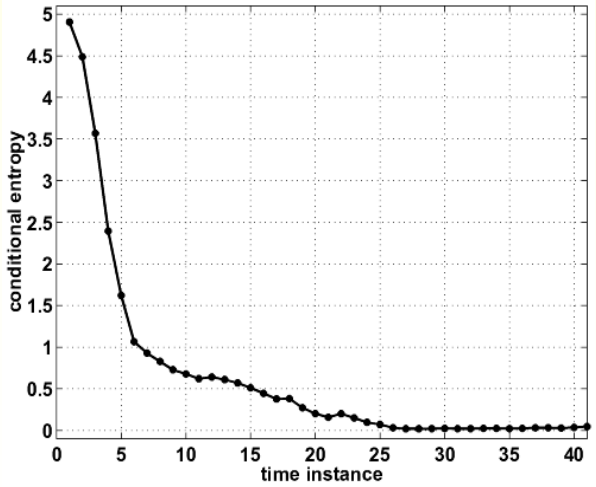
Tracking-and-recognition result



Scale estimate



Posterior probability $p(n_t|y_{1:t})$



Entropy H of $p(n_t|y_{1:t})$

Tracking accuracy and recognition rate

Case	Case 1	Case 2	Case 3	Case 4
Tracking Accuracy	87%	93%	100%	NA
Recognition w/in top 1	NA	83%	93%	57%
Recognition w/in top 3	NA	97%	100%	83%

Case 1: Pure tracking using a Laplacian density.

Case 2: Tracking-and-recognition using an IPS density.

Case 3: Tracking-and-recognition using a combined density.

Case 4: Tracking-then-recognition face recognition.

Dictionaries for signal and image analysis

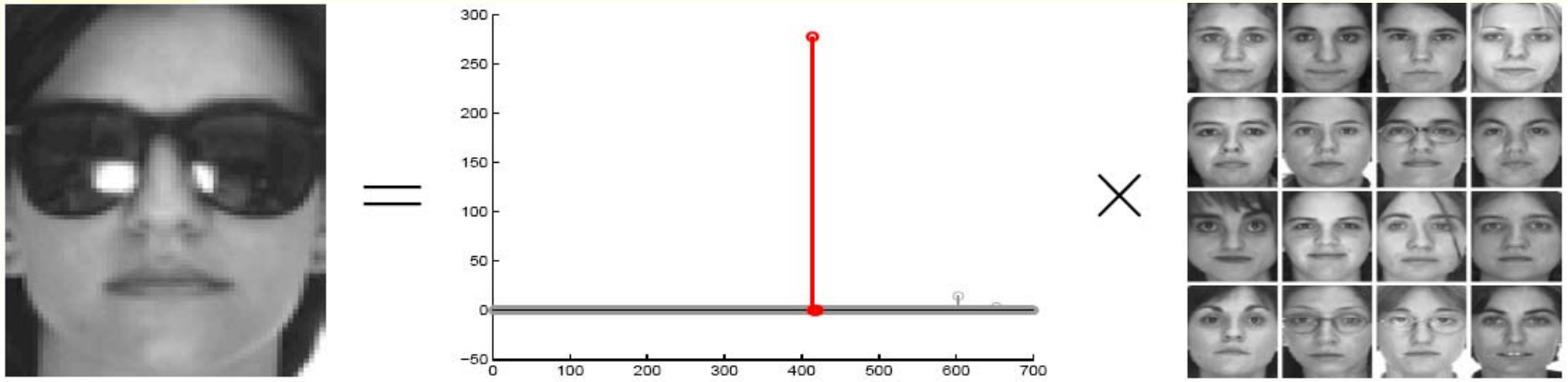
- Matching Pursuit algorithms Mallat (early 90's)
- Orthogonal matching pursuits (Pati, et al,1993, Tropp 2004)
- Saito and Coifman, 1997
- Etemad, Chellappa, 1997
- Represent signals using wavelets, wavelet packets,..
- Learning dictionary from data instead of using off-the-shelf bases. (Olshausen and Field, 1997)

Modern day dictionaries

- Represent Signals and images using signals and images!
- Sparse coding
- Allow compositional representations
- Dictionary updates
  - Batch (Method of Optimal directions)
  - K-SVD
- Dictionaries for images are more complicated
  - Need to account for pose, illumination, resolution variations.

Robust face recognition

- Automatic face recognition algorithm robust to occlusion, expressions and disguise.
- Represent the test face as a sparse linear combination of the training faces.
- Estimate the class of the test image from the sparse coefficients.
- Can identify and reject non face images.
- Wright et al., TPAMI 2009, CVPR 2010, TPAMI 2012
- Patel, et al., TIFS, 2012.



Test image

Sparse coefficients

Training images

Basic formulation

- Assume L classes and n images per class in gallery.
- The training images of the kth class is represented as

$$\mathbf{D}_k = [\mathbf{x}_{k1}, ..., \mathbf{x}_{kn}]$$

- Dictionary D is obtained by concatenating all the training images

$$\begin{aligned}\mathbf{D} &= [\mathbf{D}_1, ..., \mathbf{D}_L] \in \mathbb{R}^{N \times (n.L)} \\ &= [\mathbf{x}_{11}, ..., \mathbf{x}_{1n} | \mathbf{x}_{21}, ..., \mathbf{x}_{2n} | ..... | \mathbf{x}_{L1}, ..., \mathbf{x}_{Ln}]\end{aligned}$$

- The unknown test vector can be represented as a linear combination of the training images as

$$\mathbf{y} = \sum_{i=1}^{L} \sum_{j=1}^{n} \alpha_{ij} \mathbf{x}_{ij}$$

Basic formulation - 2

- In a more compact form $\mathbf{y} = \mathbf{D}\alpha$.

$$\alpha = [\alpha_{11}, ..., \alpha_{1n} | \alpha_{21}, ..., \alpha_{2n} | ..... | \alpha_{L1}, ..., \alpha_{Ln}]^T$$

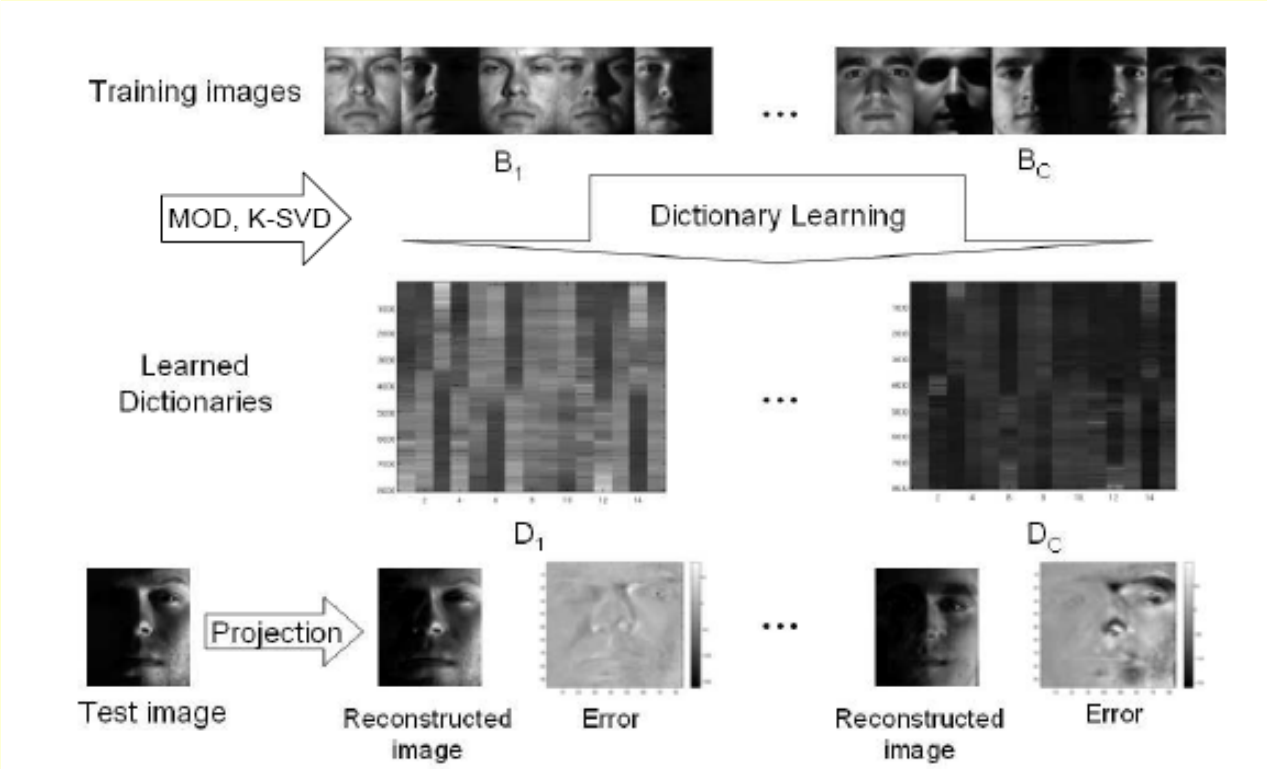
- We make the assumption that the test image can be written as a linear combination of the training images of the correct class alone.
- So the coefficient vector α is sparse.
- Hence α can be recovered by Basis Pursuit as

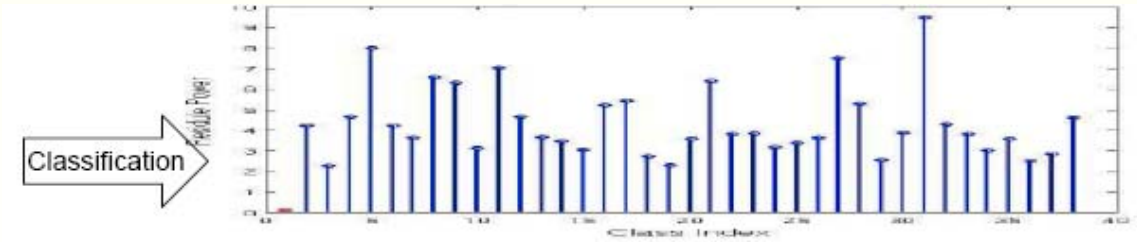
$$\hat{\alpha} = \arg \min_{\alpha' \in \mathbb{R}^N} \| \alpha' \|_1 \quad \text{subject to} \quad \mathbf{y} = \mathbf{D}\alpha'$$

Selection and recognition algorithm

- Given the gallery, construct the dictionary D by arranging the training images as its columns.
- Using the test image, by Basis Pursuit, obtain the coefficient vector α.
- Obtain the Sparsity Concentration Index (SCI).
- Compare SCI with a threshold to reject the poorly acquired images.
- Find the reconstruction error while representing the test image with coefficients of each class separately.
- Select the class giving the minimum reconstruction error.

Dictionary-based face recognition





Classification result among several classes

Learning dictionaries – K-SVD

Objective: Find the best dictionary to represent the samples $\mathbf{B} = [\mathbf{x}_1, \cdots, \mathbf{x}_m]$ as sparse compositions, by solving the following optimization problem:

$$\arg\min_{\mathbf{D},\Gamma} \|\mathbf{B} - \mathbf{D}\Gamma\|_F^2 \text{ subject to } \forall i \ \|\gamma_i\|_0 \leq T_0.$$

Input: Initial dictionary $\mathbf{D}^{(0)} \in \mathbb{R}^{N \times P}$, with normalized columns, signal matrix $\mathbf{B} = [\mathbf{x}_1, \cdots, \mathbf{x}_m]$ and sparsity level T_0.

Output: Trained dictionary $\mathbf{D}$ and sparse representation matrix Γ.

Procedure:

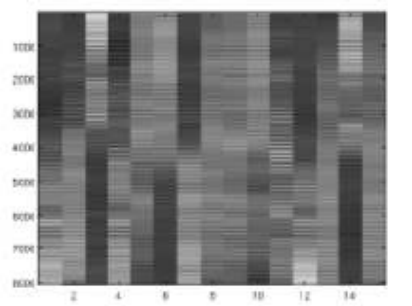
Set $J = 1$. Repeat until convergence:

- *Sparse coding stage*: Use any pursuit algorithm to compute the sparse representation vectors γ_i for each signal $[\mathbf{x}_1, \cdots, \mathbf{x}_m]$.
- *Dictionary update stage*: For each column $k = 1, \cdots, P$in $\mathbf{D}^{(J-1)}$ update by
  - Define the group of examples that use this atom, $\omega_k = \{i | 1 \leq i \leq P, \gamma_T^k(i) \neq 0\}$.
  - Compute the overall representation error matrix, $\mathbf{E}_k$, by
  $$\mathbf{E}_k = \mathbf{B} - \sum_{j \neq k} \mathbf{d}_j \gamma_T^j.$$
  - Restrict $\mathbf{E}_k$ by choosing only the columns corresponding to ω_k and obtain $\mathbf{E}_k^R$.
  - Apply SVD decomposition $\mathbf{E}_k^R = \mathbf{U}\Delta\mathbf{V}^T$. Select the updated dictionary column $\hat{\mathbf{d}}_k$to be the first column of $\mathbf{U}$. Update the coefficient vector γ_R^k to be the first column of $\mathbf{V}$ multiplied by $\Delta(1,1)$.
- Set $J = J + 1$.

Given a set of training examples, $\mathbf{B} = [\mathbf{x}_1, \cdots, \mathbf{x}_m]$, find the dictionary that leads to the best representation for each member in this set, under strict sparsity constraints.

Training faces

K-SVD

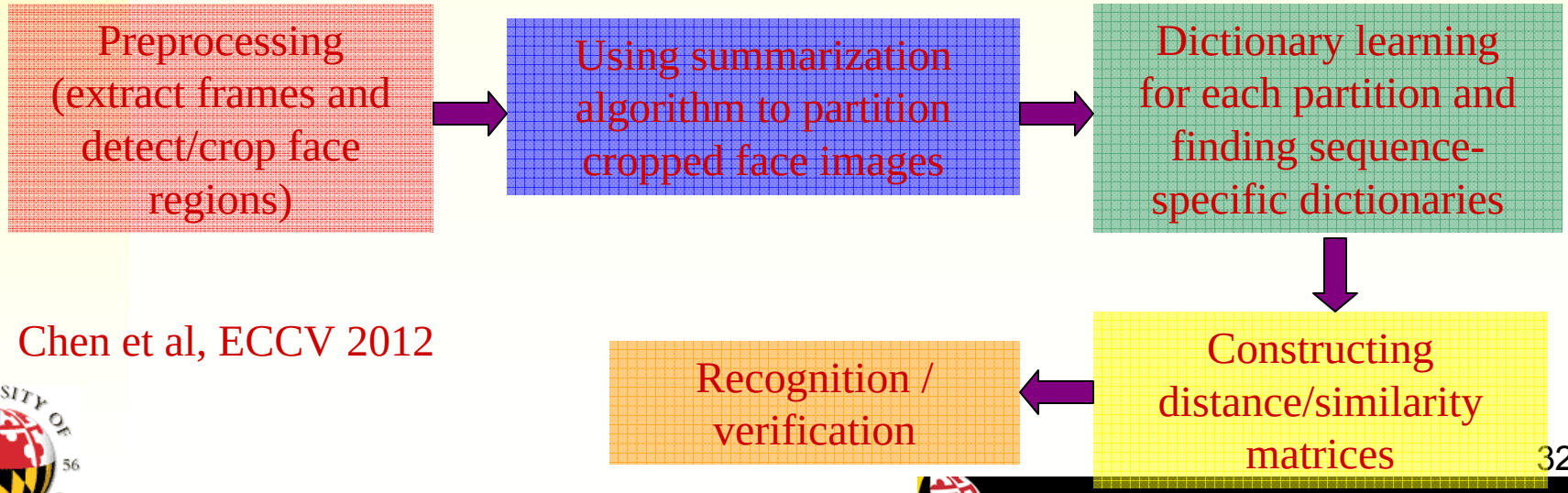


Learned dictionary

M. Aharon, M. Elad, and A. M. Bruckstein, 2006

Video dictionaries for face recognition

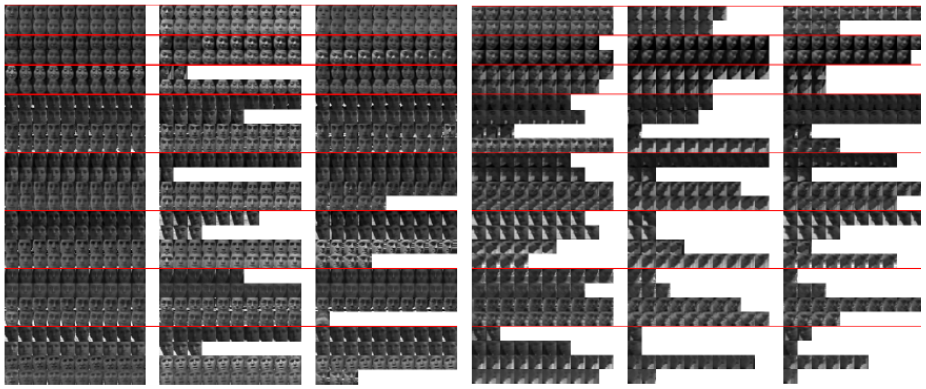
- Video-based face recognition/verification has recently drawn people's great attention due to its broad applications to video surveillance for security purposes.
- Dictionary method has recently become a very power tool for (still-image) face recognition
- Video dictionary-based face recognition steps:



Chen et al, ECCV 2012

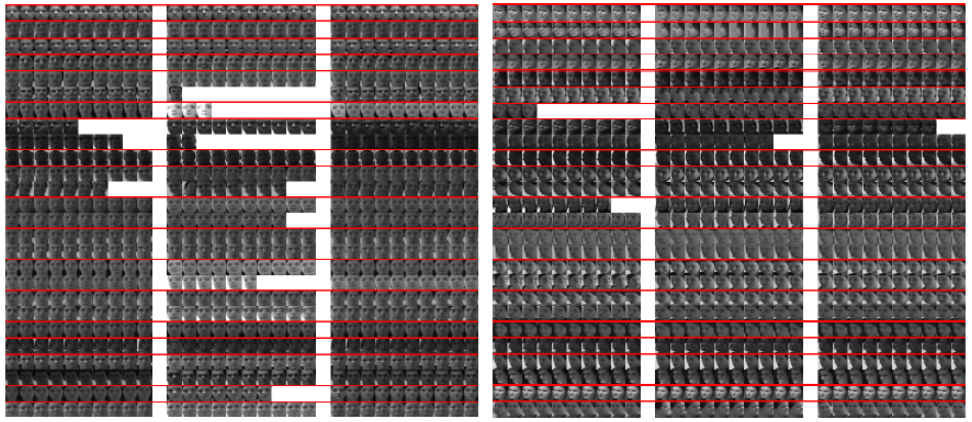
Video preprocessing/partitioning

- Given a video sequence, extract all frames from it, from which the human face regions are then detected and cropped.
- Based on video summarized algorithm [1], partition cropped face images into *K* partitions.
  - Different partitions exhibit different pose/lighting conditions.



(a) MBGC Notre Dame frontal face partitions (b) MBGC Notre Dame profile face partitions

Figure 3. MBGC Notre Dame partition results



(a) FOCS UT-Dallas walking partitions (b) FOCS UT-Dallas activity partitions

Figure 9. FOCS UT-Dallas partitions

[1] N. Shroff, P. Turaga, and R. Chellappa, "Video Precis: High lighting diverse aspects of videos," *IEEE Transactions on Multimedia*, 2010.

Dictionary learning
(build sequence-specific dictionaries)

- Let $\mathbf{G}_{j,k}^{i}$ be the gallery matrix of the *k*-th partition of the *j*-th video sequence of subject *i*.
- Given $\mathbf{G}_{j,k}^{i}$, use K-SVD [2] algorithm to build a (partition level) sub-dictionary $\hat{\mathbf{D}}_{j,k}^{i}$ such that

$$(\hat{\mathbf{D}}_{j,k}^{i}, \hat{\mathbf{\Gamma}}_{j,k}^{i}) = \underset{\mathbf{D}_{j,k}^{i}, \mathbf{\Gamma}_{j,k}^{i}}{\operatorname{argmin}} \|\mathbf{G}_{j,k}^{i} - \mathbf{D}_{j,k}^{i}\Gamma_{j,k}^{i}\|_F^2 \text{ s.t.}$$
$$\|\gamma_l\|_0 \leq T_0, \forall l,$$

- Concatenate the (partition-level) sub-dictionaries to form a sequence-specific dictionary

$$\mathbf{D}_{j}^{i} = [\mathbf{D}_{j,1}^{i}\ \mathbf{D}_{j,2}^{i} \ldots \mathbf{D}_{j,k}^{i}]$$

[2] M. Aharon, M. Elad and A. M. Bruckstein, "The K-SVD: an algorithm for designing of overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311-4322, 2006

Recognition/identification

- Given the *m*-th query video sequence $\mathbf{Q}^{(m)} = \bigcup_{k=1}^{K} \mathbf{Q}_k^{(m)}$
- We generate the partition $\mathbf{Q}_k^{(m)}$ as $\mathbf{Q}_k^{(m)} = [\mathbf{q}_{k,1}^{(m)} \ \mathbf{q}_{k,2}^{(m)} \ \cdots \ \mathbf{q}_{k,n_k}^{(m)}]$
- The distance $\mathbf{R}^{(m,p)}$, between $\mathbf{Q}^{(m)}$ and $\mathbf{D}_{(p)}$ (i.e. dictionary of the *p*-th video sequence) is calculated as

$$\mathbf{R}^{(m,p)} = \min_{k \in \{1,2,\ldots,K\}} \mathbf{R}_k^{(m,p)}$$

where

$$\mathbf{R}_k^{(m,p)} \triangleq \frac{1}{n_k} \left(\sum_{l=1}^{n_k} \| \mathbf{q}_{k,l}^{(m)} - \mathbf{D}_{(p)} \mathbf{D}_{(p)}^{\dagger} \mathbf{q}_{k,l}^{(m)} \|_2 \right)$$

- We select the best matched $\mathbf{D}_{(p^*)}$ with $\mathbf{Q}^{(m)}$ such that

$$p^* = \underset{p \in \{1,2,\ldots,P\}}{\operatorname{argmax}} \ \mathbf{R}^{(m,p)}$$

Video-based face recognition using dictionaries

- Summary of the proposed algorithm

Preprocessing/dictionary learning stages:

- Given the j-th video sequence of subject i, extract all frames from it and detect/crop face regions. All detected and cropped face images form a set S_j^i.
- Partition S_j^i into K partitions. Augment each partition by adding artificial gitters and obtain resulting augmented gallery matrix from the k-th partition, $\mathbf{G}_{j,k}^i, \forall k = 1, 2, ..., K$.
- Use K-SVD algorithm to learn the partition-specific sub-dictionary $\mathbf{D}_{j,k}^i, \forall k = 1, 2, ..., K$. Construct sequence-specific dictionary $\mathbf{D}_j^i$.

Recognition/verification stages:

- Partition the m-th query video sequence $\mathbf{Q}^{(m)} = \bigcup_{k=1}^{K} \mathbf{Q}_k^{(m)}$, where $\mathbf{Q}_k^{(m)} = [\mathbf{q}_{k,1}^{(m)}\ \mathbf{q}_{k,2}^{(m)}\ ...\ \mathbf{q}_{k,n_k}^{(m)}]$.
- Find the distance $\mathbf{R}^{(m,p)}$ between $\mathbf{Q}^{(m)}$ and $\mathbf{D}_{(p)}$.
- Use subject-sequence correspondence $\boldsymbol{m}(\cdot)$ to make the final decision.
- Use $\mathbf{R}^{(m,p)}$ to construct the distance matrix, from which recognition rates and ROC curves can be obtained.

Experiments on MBGC (Notre Dame) videos version 1

- 397 walking (frontal-face) videos: 198 SDs + 199 HDs
- 371 activity (profile-face) videos: 185 SDs + 186 HDs

(a) MBGC frontal face scenario 1

(b) MBGC frontal face scenario 2

(c) MBGC frontal face scenario 3

(d) MBGC frontal face scenario 4

Figure 1. MBGC frontal face scenarios

(a) MBGC profile face scenario 1

(b) MBGC profile face scenario 2

(c) MBGC profile face scenario 3

(d) MBGC profile face scenario 4

Figure 2. MBGC profile face scenarios

Experiments - Recognition results

- Leave one out test on MBGC Notre Dame walking videos
  - Three subsets are selected for testing: S2 (subjects with at least 2 video sequences), S3 (subjects with at least 3 video sequences) and S4 (subjects with at least 4 video sequences)

MBGC Notre Dame walking videos	Arc-length Metric	Procrustes Metric	Kernel Density	Wrapped Gaussian Common Pole [8]	Dictionary Method
S2 (143 subjects, 395 videos): leave one out	38.48	43.79	39.74	63.79	**79.09**
S3 (55 subjects, 219 videos): leave one out	48.85	53.88	50.22	74.88	**79.91**
S4 (54 subjects, 216 videos): leave one out	48.61	53.70	50.46	75	**80.09**

Table 2. Recognition rates (%) of leave-one-out testing experiments on MBGC Notre Dame walking videos

[8] P. Turaga, A. Veeraraghavan, A. Srivastava, and R. Chellappa, "Statistical computations on grassmann and stiefel manifolds for image and video-based recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.

Experiments - Verification results

- ROCs of "SD v.s. HD" and "HD v.s. SD" tests on MBGC Notre Dame walking videos
  - SD (standard definition): 720x480, HD (high definition): 1440x1080

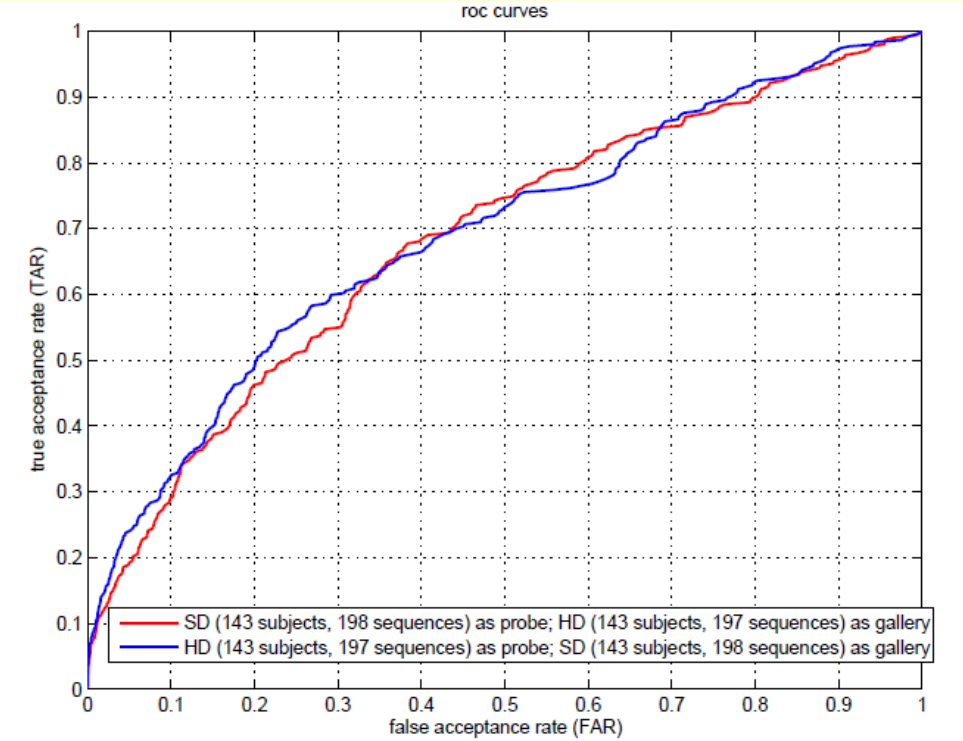
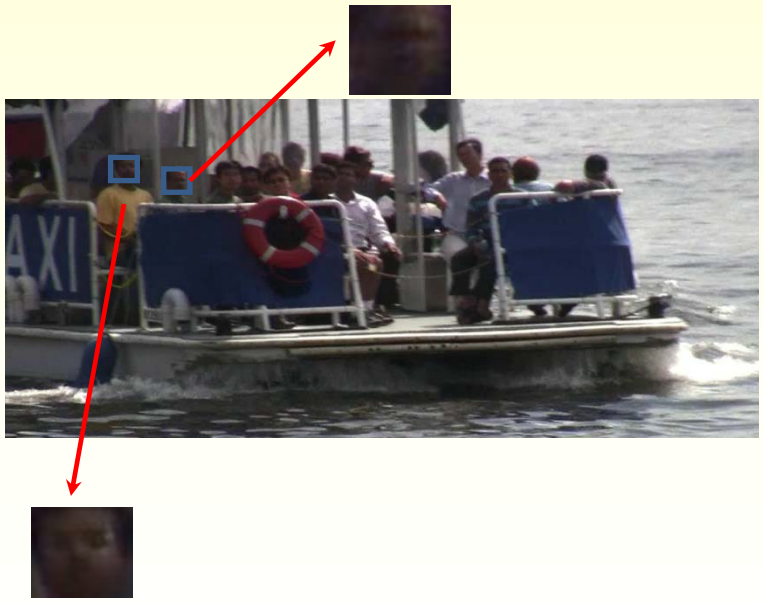


Figure 4. ROCs of SD v.s. HD and HD v.s. SD testing experiments on MBGC frontal (walking) videos using Dictionary method

Face association

- Applications: personal album labeling, video archiving, face database generation.
- Effective for video-based face recognition.



Successive Detection Vs. Tracking

- Guarantee temporal coherence, automatic face association.
- Prone to errors caused by low resolution, long range sensor, Uncontrolled lighting, pose, occlusion etc.
- Shaky camera, motion blur, stabilization fails.

Successive detection Vs. tracking

- Stable result.
- Large numbers of false positives and false negatives.

Face detection: AdaBoost [Viola-Jones, 2004]
Body detection: Deformable-model based [Felzenszwalb, et al, 2008]
Fusion of results at two resolutions
False detection removed by Multi-scale LBP and SVMs

Problem formulation

- Each detected face candidate is a node: $y_n,\ n = 1, 2, ..., N$
- Each state corresponding to a previously recorded face:

$$\mathfrak{S} = \{1, 2, ..., S\}$$

- A null state to account for unseen faces or false detections:

$$\mathfrak{S}^+ = \{0, 1, 2, ..., S\}$$

- The overall configuration space: $\mathfrak{Y} = \underbrace{\mathfrak{S}^+ \times \mathfrak{S}^+ \times ... \times \mathfrak{S}^+}_{N}$

- Solve for the optimal configuration of nodes $\mathbf{Y}_t$ in the augmented state space $\mathfrak{S}_t^+$
- All-connected graph, time-varying nodes and state space.

Du and Chellappa, ECCV 2012

General conditional random field definition

- A conditional random field is a graphical structure of (**x**,**y**) such that p(**y**|**x**) has the following factorization form ([Lafferty et al. ICML 01]):

c is a clique of the graph

$$p(\mathbf{y} \mid \mathbf{x}) = \frac{\prod_{c \in C(G)} \phi_c(\mathbf{y}_c, \mathbf{x})}{Z(\mathbf{x})}$$

Potential ϕ_C can use all of **y**, not only $\mathbf{x}_C$

Partition function Z(**x**) is the normalization factor, so that p(**y**|**x**) sums to one:

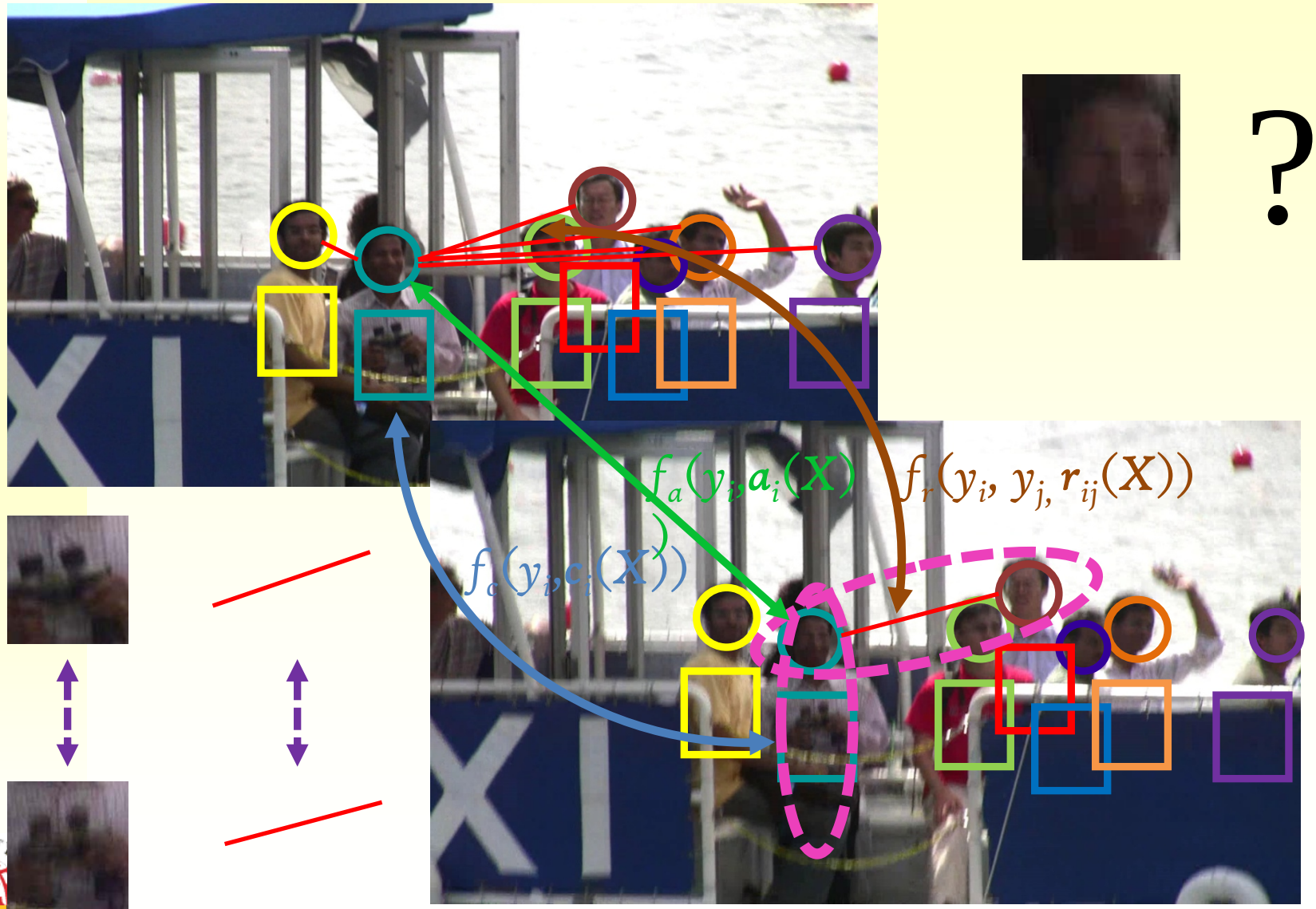
$$Z(\mathbf{x}) = \sum_{\mathbf{y}} \prod_{c \in C(G)} \phi_c(\mathbf{y}_c, \mathbf{x})$$

Problem formulation

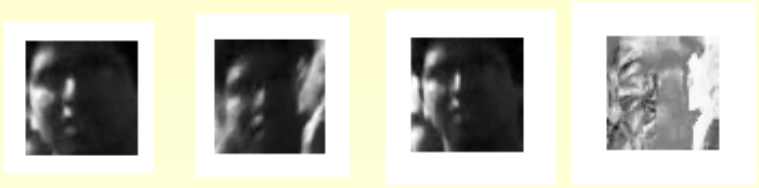
$$\log p(\mathbf{Y}|\mathbf{X},\theta) = -\log Z(\mathbf{X},\theta) + \sum_i \theta_a f_a(y_i, \mathbf{a}_i(\mathbf{X})) + \sum_i \theta_c f_c(y_i, \mathbf{c}_i(\mathbf{X})) + \sum_{(i,j)\in E} \theta_r f_r(y_i, y_j, \mathbf{r}_{ij}(\mathbf{X})) + \sum_{(i,j)\in E} \theta_u f_u(y_i, y_j)$$

$f_a(y_i, \mathbf{a}_i(\mathbf{X}))$	: Face appearance
$f_c(y_i, \mathbf{c}_i(\mathbf{X}))$	: Clothing appearance
$f_r(y_i, y_j, \mathbf{r}_{ij}(\mathbf{X}))$	: Relative pose
$f_u(y_i, y_j)$	: Uniqueness constraint

Context-aided face matching



Face appearance



- Adaptive Appearance Model: A Gaussian mixture model with Stable, Wander and Lost components (Jepson et al. CVPR01)

$$p(\mathrm{a}(\mathrm{X}_t)|\mathcal{A}_{g,t-1}) = \prod_{b=1}^{B}[m_{w,t,b}p_w(a_{t,b}|a_{g,t-1,b},\sigma^2_{g,w}) + m_{s,t,b}p_s(a_{t,b}|\mu_{g,s,t,b},\sigma^2_{g,s,t,b}) + m_{l,t,b}p_l(a_{t,b})]$$

Stable Wander Lost

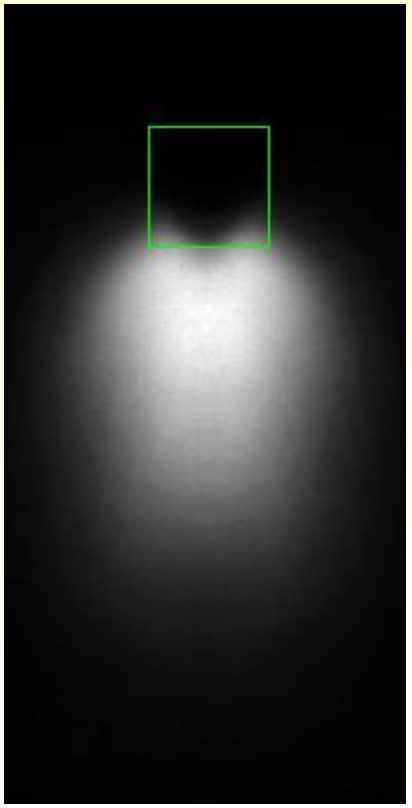
- Online EM algorithm

$$L = \sum_{T=-\infty}^{t} S_t(T)\log p(a_T|\mathcal{A}_{g,T-1}) \ , \ S_t(T) = (1-e^{-1/\tau})e^{-(t-T)/\tau}$$

Clothing appearance

- Color histogram in YCbCr Space, Chi-square distance

H3D database [L. Bourdev and J. Malik 2009]
http://www.eecs.berkeley.edu/~lbourdev/poselets



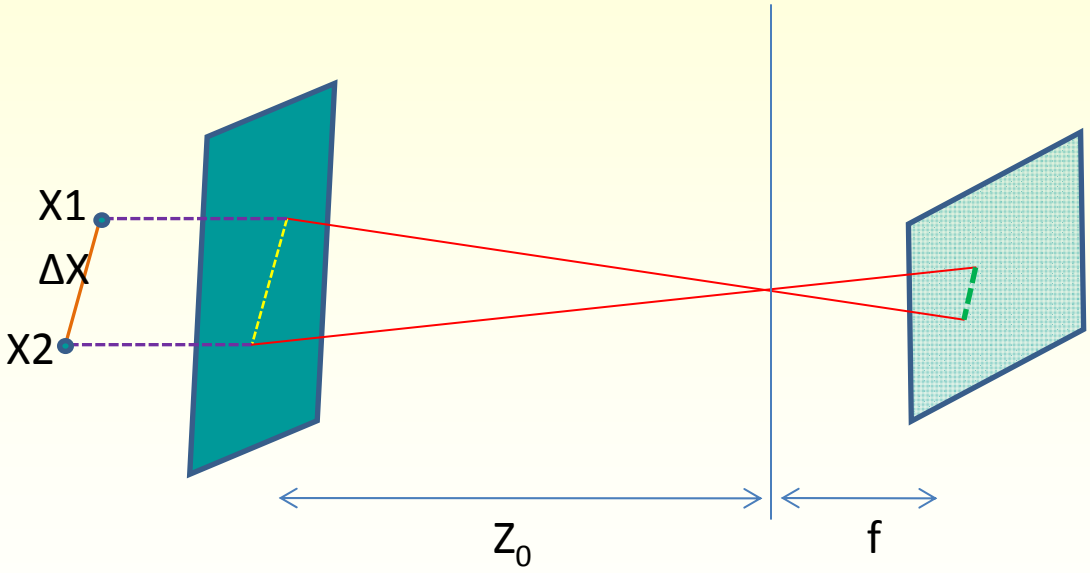
Hard constraint

- No person can appear twice in the same frame!

$$f_u(y_i, y_j) = \begin{cases} -\inf & \text{if } y_i = y_j \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

Relative pose

- Based on the weak-perspective model (because of the long capture distance)



Relative pose

Assume that the camera undergoes a large shift so that the principle point moves from (0, *0) to (px, py). Also assume there is a zooming effect caused* by change in focal length from *f to αf. Suppose a scene point P1 moves a short distance from* (*X1, Y1,Z1) to (X1 +δX, Y1 +δY ,Z1 +δZ) and another scene point P2 stands still at (X2, Y2, Z2).* define a scale-normalized relative distance change (in *x direction) for P1 and P2's images as follows:*

$$
\begin{aligned}
D(\Delta_x, \Delta_{x'}, \alpha) &= \frac{1}{\alpha f}\Delta_{x'} - \frac{1}{f}\Delta_x \\
&= \frac{1}{\alpha f}[(\alpha f \frac{X_1 + \delta_X}{Z_0} + p_x) - (\alpha f \frac{X_2}{Z_0} + p_x)] - \frac{1}{f}(f\frac{X_1}{Z_0} - f\frac{X_2}{Z_0}) = \frac{\delta_X}{Z_0}
\end{aligned}
$$

$$
\mathbf{r}_{ij}(\mathbf{X}_t) = (\Delta\mu_{i,j,t} = \mu_{j,t} - \mu_{i,t}, \Delta\nu_{i,j,t} = \nu_{j,t} - \nu_{i,t}, \Delta\omega_{i,j,t} = \frac{\omega_{j,t}}{\omega_{i,t}})
$$

$$
\begin{aligned}
f_r(y_i, y_j, \mathbf{r}_{ij}(\mathbf{X}_t), \mathbf{r}_{y_i y_j}(\mathbf{X}_{t-1}), \theta_r) &= \theta_{r1} \log Lap(\frac{1}{\omega_{i,t}}\Delta\mu_{i,j,t} - \frac{1}{\omega_{y_i,t-1}}\Delta\mu_{y_i,y_j,t-1}|m_{r1}, b_{r1}) \\
&+ \theta_{r2} \log Lap(\frac{1}{\omega_{i,t}}\Delta\nu_{i,j,t} - \frac{1}{\omega_{y_i,t-1}}\Delta\nu_{y_i,y_j,t-1}|m_{r2}, \sigma^2_{\Delta\nu}) \\
&+ \theta_{r3} \log Lap(\Delta\omega_{i,j,t} - \Delta\omega_{i,j,t-1}|m_{r3}, b_{r3})
\end{aligned}
$$

Dynamic CRF

- Number of nodes and states varies with subjects entering or leaving the scene.
- Feature function implicitly changes via appearance adaption.
- The “NULL” state: Modeling unseen subjects or outliers.

$$Z(\mathbf{X}) = \mathbf{f} = [f(y_i = 1, \mathbf{X}), f(y_i = 2, \mathbf{X}), ..., f(y_i = S, \mathbf{X})]^T$$

Logistic Regression

$$Z'(\mathbf{f}) = \mathbf{f}' = [f'_0, f'_1, ..., f'_S] \ , \ f'_s = \frac{e^{\mathbf{w}_s^T \phi(\mathbf{f})}}{\sum_{m=0}^{S} e^{\mathbf{w}_m^T \phi(\mathbf{f})}}$$

CRF parameter estimation: Algorithm

Input: N labeled training samples$\{\mathbf{X}^{(n)}, \mathbf{Y}^{(n)}\}$

Output: Optimal parameters θ^*

while $L_t - L_{t-1} > threshold$ **do**

  for $n = 1 \rightarrow N$ **do**

    - Calculate the un-normalized potential
      $\sum_i \sum_p \theta_{p,t-1} f_p(y_i^{(n)}, \mathbf{X}^{(n)}) + \sum_{(i,j)\in E} \sum_q \theta_{q,t-1} f_q(y_i^{(n)}, y_j^{(n)}, \mathbf{X}^{(n)})$;
    - Apply the Gibbs Sampling algorithm to draw M model samples $\{\vec{\mathbf{Y}}^{(m)}, m = 1, ..., M\}$;
    - Do inference, calculate $Z(\mathbf{X}^{(n)}, \theta_t)$ and $p(\mathbf{Y}^{(n)}|\mathbf{X}^{(n)}, \theta_t)$;

    for $p = \{a, c\}$,$q = \{r1, r2, r3\}$ **do**

      - Calculate $A_n = \sum_{m=1}^{M} p(\vec{\mathbf{Y}}^{(m)}|\mathbf{X}^{(n)}, \theta_t) \sum_i f_p(\vec{y}_i^{(m)}, \mathbf{X}^{(n)})$;
      - Calculate $B_n = \sum_{m=1}^{M} p(\vec{\mathbf{Y}}^{(m)}|\mathbf{X}^{(n)}, \theta_t) \sum_{(i,j)\in E} f_q(\vec{y}_i^{(m)}, \vec{y}_j^{(m)}, \mathbf{X}^{(n)})$;

    end

  end

  for $p = \{a, c\}$,$q = \{r1, r2, r3\}$ **do**

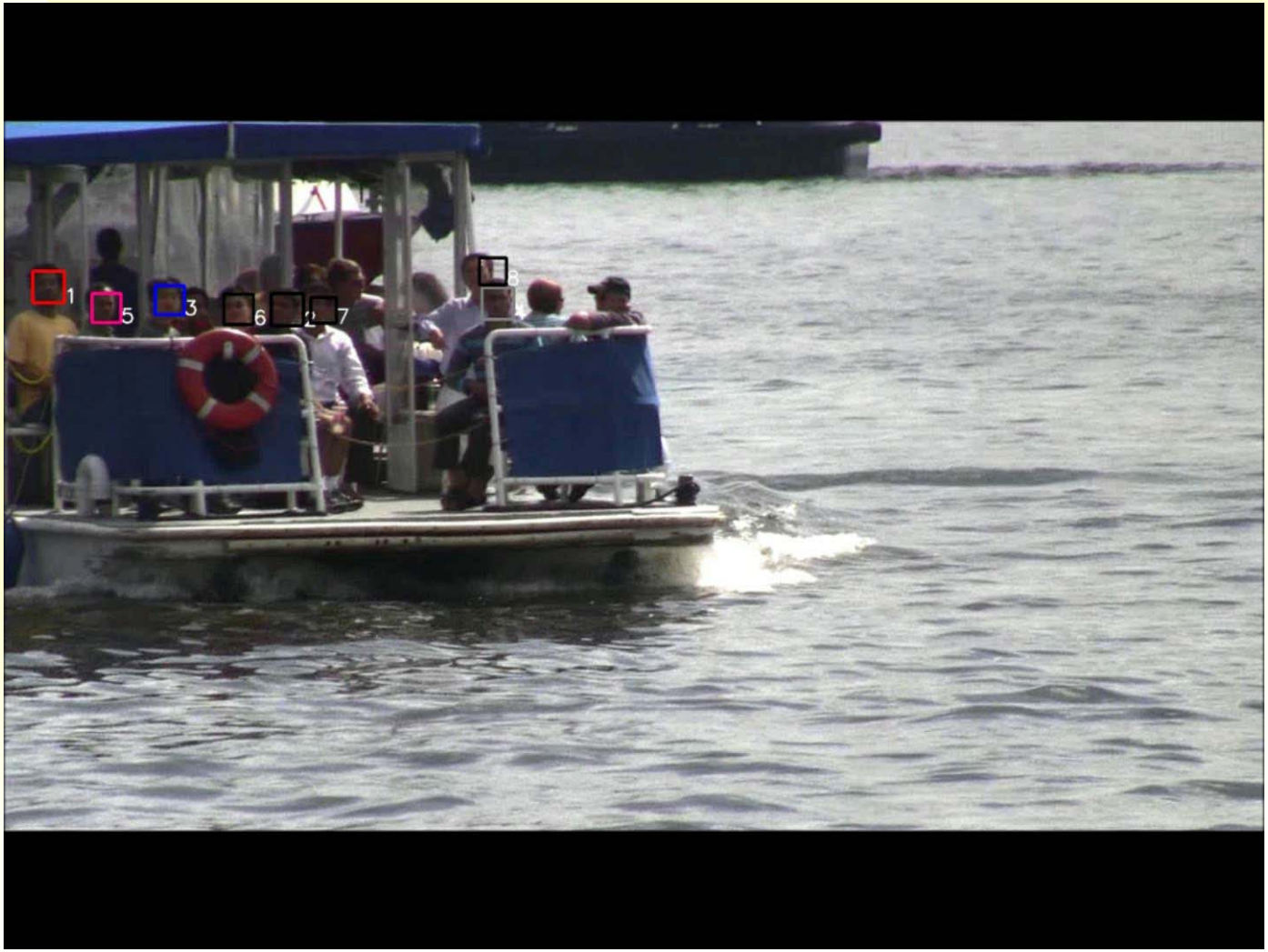
    - Evaluate $\frac{\partial L}{\partial \theta_p}$ and $\frac{\partial L}{\partial \theta_q}$
    - Update $\theta_{p,t} \rightarrow \theta_{p,t-1} - \lambda \frac{\partial L}{\partial \theta_p}$, $\theta_{q,t} \rightarrow \theta_{q,t-1} - \lambda \frac{\partial L}{\partial \theta_q}$;

  end

  Calculate L_t

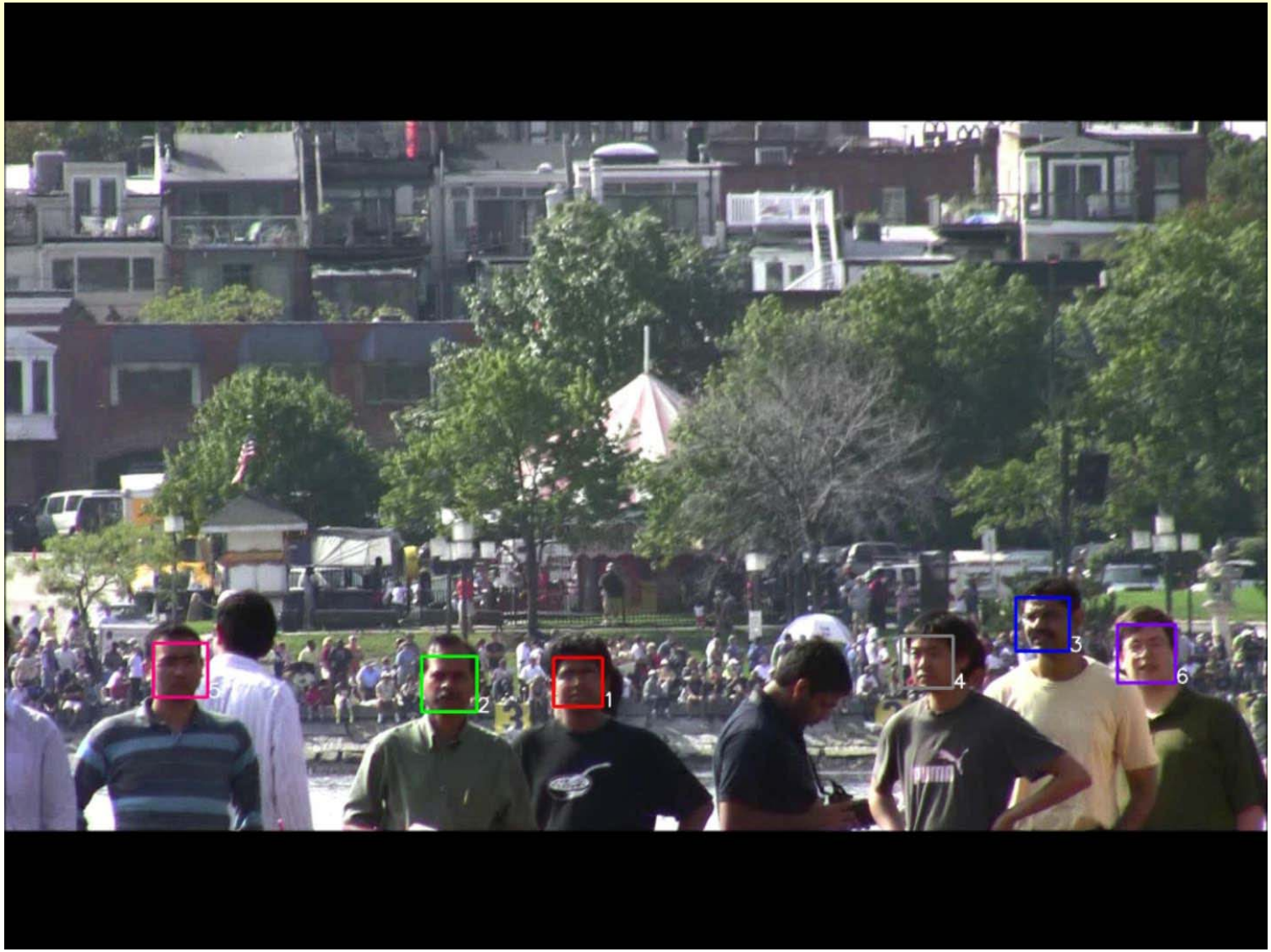
end

Result



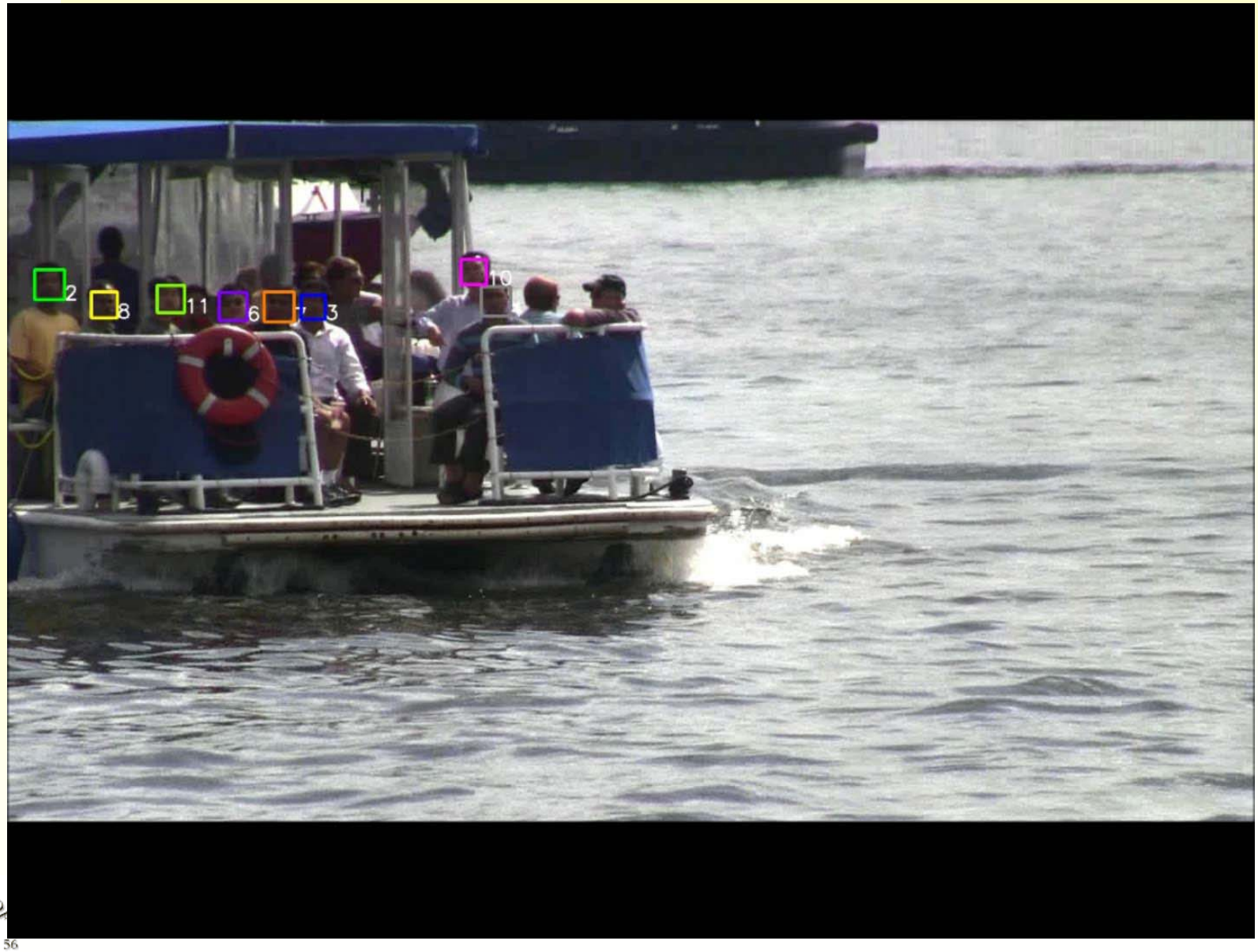
Some faces can not be detected by face detector. However, their positions can be roughly inferred from the relative pose and are marked with black bounding boxes in this video.

Result



Comparison

- Two alternative approaches:
  - Combination of face detector and face tracker + K-means clustering. Videos are first stabilized. 500 particles for each tracker.
  - Face detection + constrained K-means.
- K is given as ground truth number of face classes + 3.


2
8
11
6
3
10

Simultaneous tracking and recognition in spherical harmonics space

- Most existing multi-view face recognition algorithms follow a two-step “pose estimation – normalization” scheme.
- We hope to avoid the pose estimation or model registration by exploiting the data obtained via a camera network.

Gallery

Probe

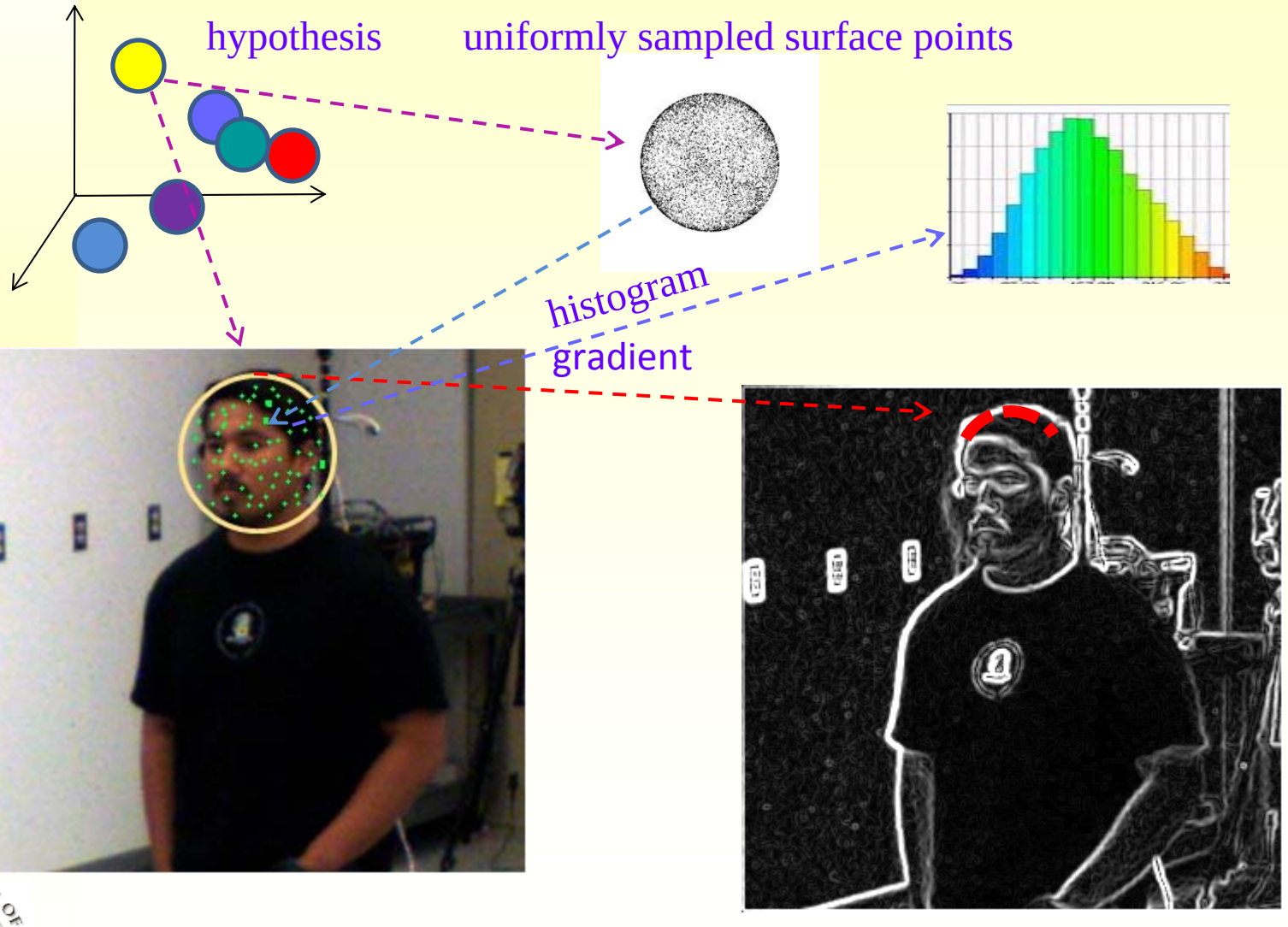
Face Matching across Pose

Multi-view Face Recognition in Camera Network

Ming Du and R. Chellappa, Under preparation

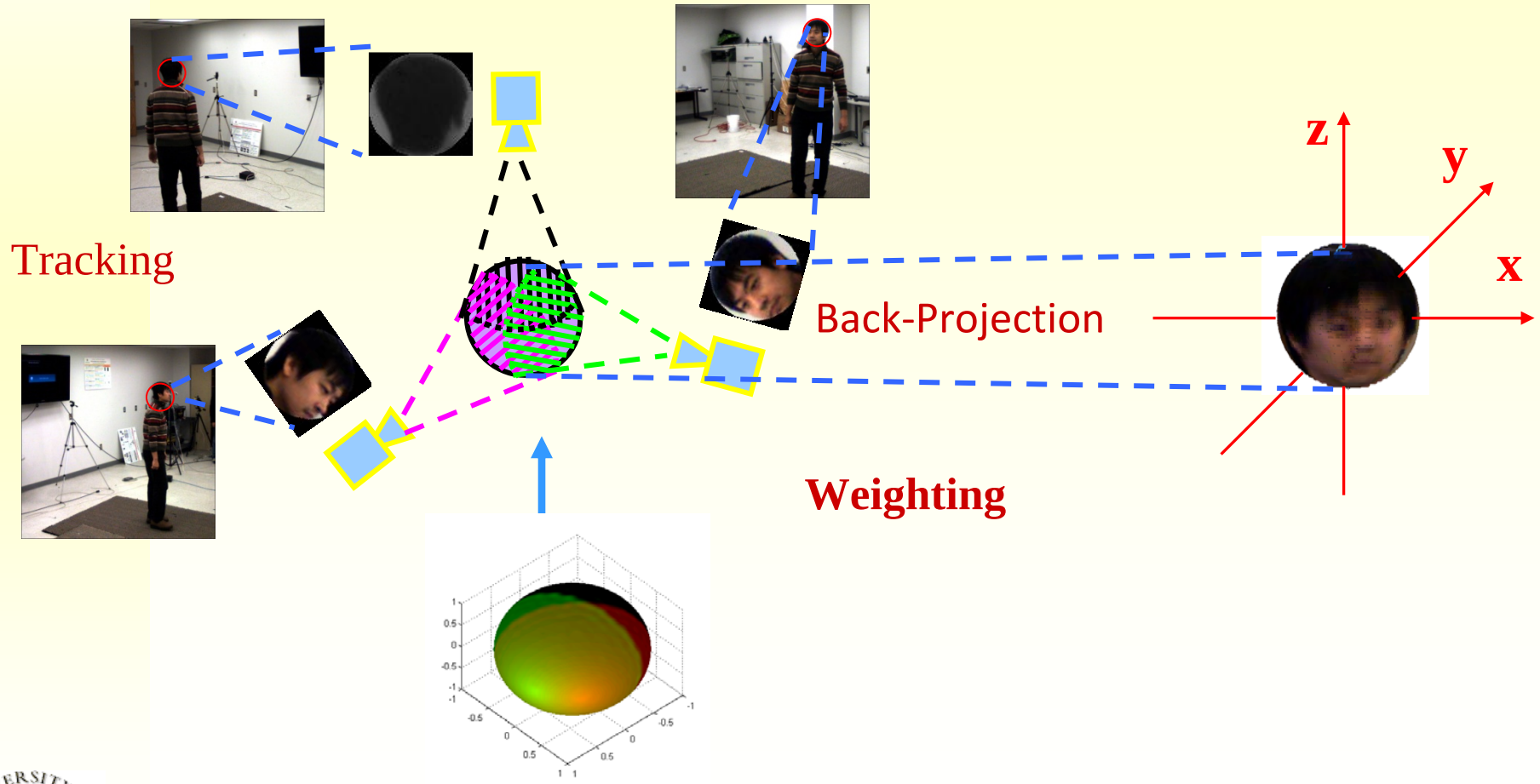
UMIACS
UNIVERSITY OF MARYLAND INSTITUTE FOR ADVANCED COMPUTER STUDIES

Tracking: Image likelihood model



Multi-camera tracking demo

Texture mapping



Spherical harmonics

- A set of orthonomal basis over sphere:

$$f(\theta,\phi)=\sum_{l=0}^{\infty}\sum_{m=-l}^{l} f_l^m Y_l^m(\theta,\phi)$$

l: degree *m* : order

- Basis functions: $Y_{lm}(\theta,\phi)=K_{lm}P_l^m(\cos\theta)e^{im\phi}$

where K_{lm} is a normalization constant and $P_l^m(x)$ is the associated Legendre functions.

l=0

l=1

l=2

… …

Spherical harmonics

- An analogue to Fourier series for spherical function.
- Decomposition equation:

$$f_l^m = \int_\theta \int_\phi f(\theta, \phi) Y_l^m(\theta, \phi) d\theta d\phi$$

- The Real Spherical Harmonics shares a lot of properties with the Spherical Harmonics.

$$Y_l^m(\theta, \phi) = \begin{cases} Y_{l0} & \text{if } m = 0 \\ \frac{1}{\sqrt{2}}(Y_{lm} + (-1)^m Y_{l,-m}) & \text{if } m > 0 \\ \frac{1}{\sqrt{2}i}(Y_{l,-m} - (-1)^m Y_{lm}) & \text{if } m < 0 \end{cases}$$

R. Basri and D. Jacobs, "Lambertian reflectance and linear subspaces," in *ICCV, vol. 2,* 2001, pp. 383–390.

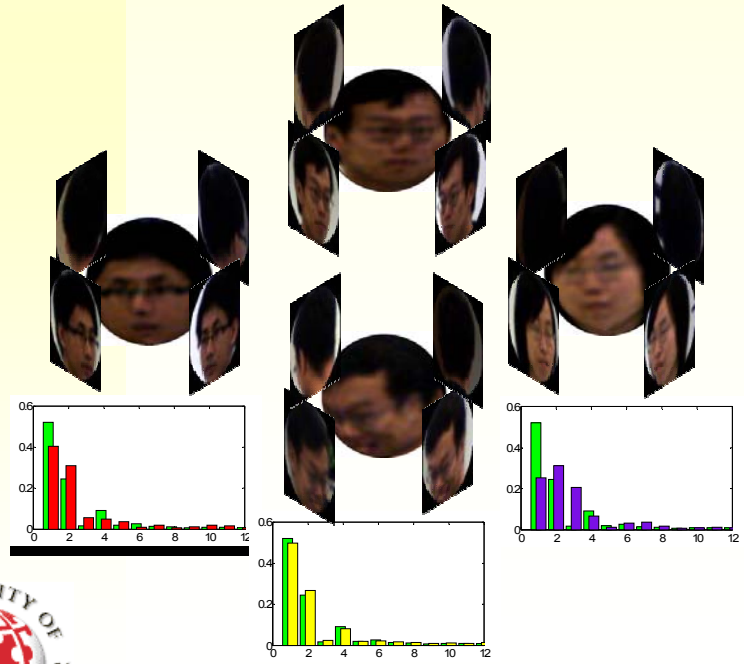
Z. Yue, W. Zhao, and R. Chellappa, "Pose-encoded spherical harmonics for face recognition and synthesis using a single image," *EURASIP Journal on Advances in Signal Process, vol.*2008, pp. 1–18, January 2008.

M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3d shape descriptors," in *Proceedings of the 2003 Eurographics/ACM SIG-GRAPH symposium on Geometry processing, June 2003, pp. 156–164.*

The SH energy dispersion feature

- **Proposition (Energy Preservation Under Rotation)** If two functions defined on S^2: $f(\theta, \phi)$ and $g(\theta, \phi)$ are related by a rotation $R \in SO(3)$, i.e. $g(\theta, \phi)=R(f(\theta, \phi))$, and their SH expansion coefficients are g_l^m and f_l^m (l =0,1, ... and $m = -l, ..l$), respectively, the following relationship exists:

$$g_l^m = \sum_{m'=-l}^{l} D_{mm'}^l f_l^{m'} \qquad \text{where} \qquad \sum_{m'=-l}^{l} (D_{mm'}^l)^2 = 1$$



Trade-off: reconstruction accuracy and computational cost.

Limiting probability distance in Reproducing kernel Hilbert space

- Image matching to image-set matching: from sample similarity to ensemble similarity [Zhou and Chellappa, PAMI 2006].
- Suppose we have two set of feature vectors (raw images, edge maps, SH coefficients etc.), think of them as samples from respective probability distributions. We try to measure the distance between two such distributions in RKHS from their gram matrix.
- The number of vectors in each ensemble may or may not be the same.
- Normality of the data is assumed. And we use the Bhattacharyya distance:
- For invertability, the covariance matrix in RKHS is approximated in such a way that the top eigenvalues and eigenvectors are preserved and the matrix is regularized.

$$J_B(\mathrm{p}_1, \mathrm{p}_2) = \tfrac{1}{8}(\mu_1 - \mu_2)^{\mathrm{T}}[\tfrac{1}{2}(\Sigma_1 + \Sigma_2)]^{-1}(\mu_1 - \mu_2) + \tfrac{1}{2}\log \frac{|\frac{1}{2}(\Sigma_1+\Sigma_2)|}{|\Sigma_1|^{1/2}|\Sigma_2|^{1/2}}$$

- Limiting probability distance is calculated when the covariance regularization parameter $\rho \rightarrow 0$.

Experiments

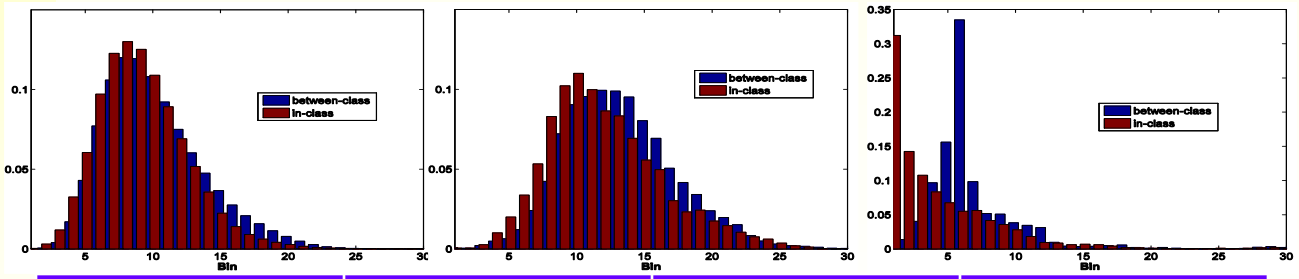
- 40 Subjects enrolled. Each subject has 1 gallery video and most subjects have 2 probe videos.
- 4-5 calibrated cameras. Indoor environment.
- 3 sessions: Intervals in-between are up to six months.

Feature comparison

- Image-based classification: A subset of the video frames.

Features	NN	KDE	SVM-Linear	SVM-RBF
LPP	56.1%	42.7%	58.8%	65.9%
LDA	51.3%	34.8%	40.6%	47.4%
SH+PCA	40.7%	36.4%	39.3%	52.2%
SH Energy	65.3%	65.1%	79.0%	87.3%

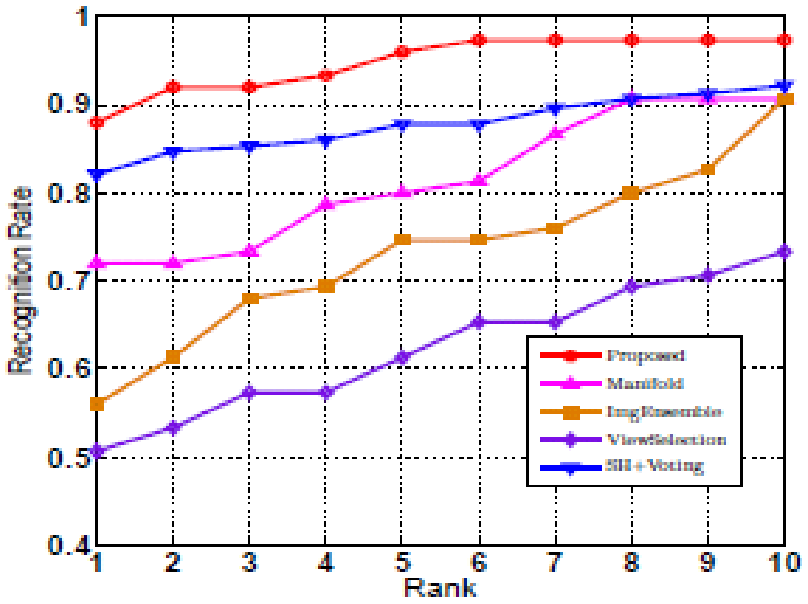
- Feature Discrimination Power: in-class and between-class feature distance



LPP	LDA	SH+PCA	SH Energy
0.3511	0.2709	0.2866	1.3141

Video recognition results

- Comparison with
  - Appearance manifold algorithm (Lee et al. CVIU 05)
  - Image ensemble matching algorithm based on limiting probability distance in RKHS (Zhou et al. PAMI 06)
  - View selection algorithm (SVM
  - SH + majority voting.



Joint Albedo Estimation and Pose Tracking from Video

Sima Taheri, A. Sankaranarayanan and R. Chellappa, ECCV 2012

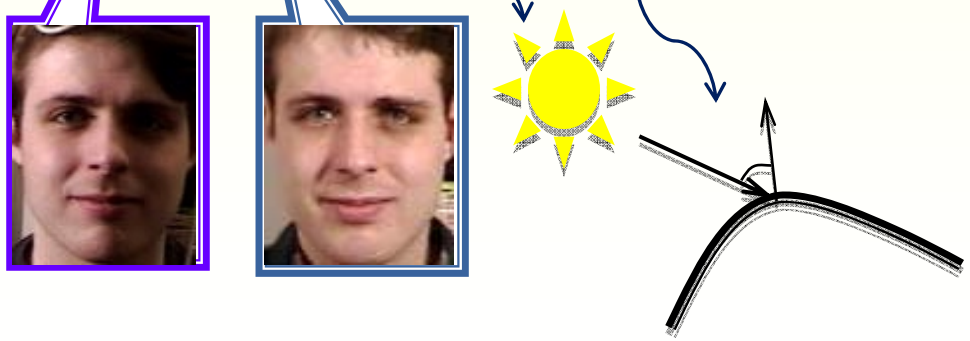
UMIACS
UNIVERSITY OF MARYLAND INSTITUTE FOR ADVANCED COMPUTER STUDIES

Motivation

- Much of the variations in the visual appearance of an object arise from effects of illumination and pose.
- The presence of invariants can go a long way in the development of robust computer vision algorithms.
- **Albedo** is the illumination-invariant property of the Lambertian reflectance surfaces.

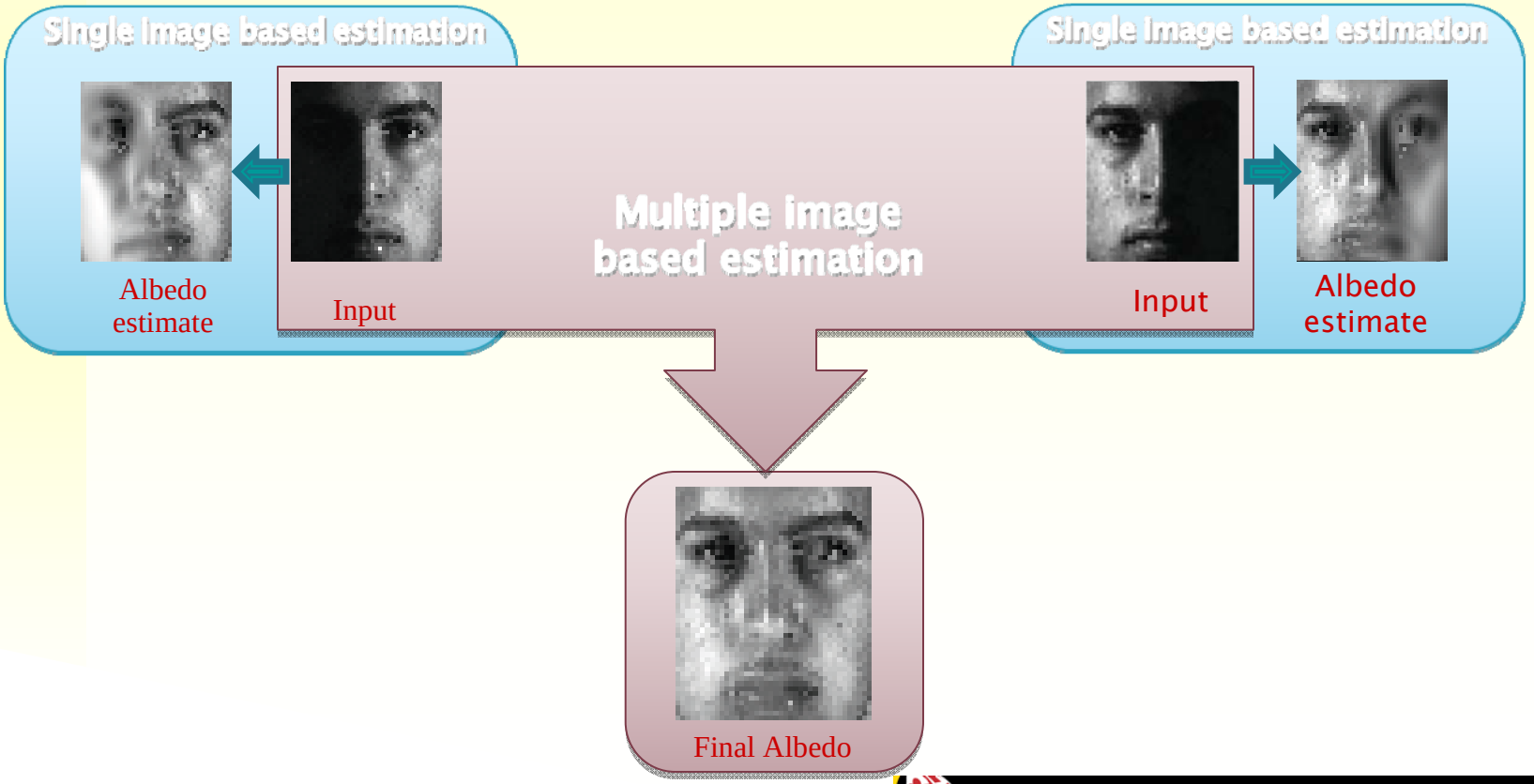
Lambert's Law:

$$I(p_i) = \rho_i \int l(\mathbf{u}_l)\ \max(\mathbf{u}_l.\mathbf{n}_i, 0)\, d\mathbf{u}_l$$



Motivation

- Albedo estimation
  - Single image vs. Multiple image



Spherical harmonic subspace

- Image representation using linear combination of its spherical harmonic basis images
  - The basis introduces an illumination-invariant representation for the object

$$I \approx \rho_i \sum_{n=0}^{2} \sum_{m=-n}^{n} l_{nm} \alpha_n y_{nm}(\mathbf{n}_i)$$

$$\left\{ \mathrm{Y}_{nm} = \{\alpha_n y_{nm}(n_i)\}_{i=1}^{i=d} \in \Re^d \mid n = 0,1,2; m = -n,\ldots,n \right\}$$

α_n : Lambert's kernel Coefficients

$$\max(\mathbf{u}_l.\mathbf{n}, 0)$$

l_{nm} : Illumination coefficients

$$l(\mathbf{u}_l) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} l_{nm} y_{nm}(\mathbf{u}_l)$$

$y_{nm}(\mathbf{n}_i) \rightarrow$ *n*th order spherical harmonic at a Point with surface normal n_i

$$\mathbf{Y} = [Y_{00},\ldots,Y_{22}]$$

$$I = \mathrm{diag}(\rho)\mathbf{YL} + \upsilon$$

Albedo estimation using Kalman filter

- Given the albedo estimate at frame/time *t-1*, characterized as $\{\mu_{\rho,t-1}, \Sigma_{\rho,t-1}\}$ we want to update $P(\rho \mid \mathcal{Z}^t, \Theta)$ knowing the pose information.

frames: $\mathcal{Z}^t = \{z_1, \ldots, z_t\}$

pose: $\Theta^{t-1} = \{\theta_1, \ldots, \theta_{t-1}\}$

$$I = \text{diag}(\rho)\mathbf{YL} + \nu \quad \Rightarrow \quad \text{Kalman Filter}$$

Pose estimation using Particle filter

- Head pose is unknown in the input video
- So the pose estimation problem is formulated as estimating $P(\theta_t \mid \rho, \mathcal{Z}^t, \Theta^{t-1})$. The albedo is known.

$$I = \text{diag}(\rho)\mathbf{Y}(\theta_t)\mathbf{L} + \nu \quad \Rightarrow \quad \text{Particle Filter}$$

Head orientation vs. Illumination direction

- To remove the problem of recalculating the basis function at each new pose/frame

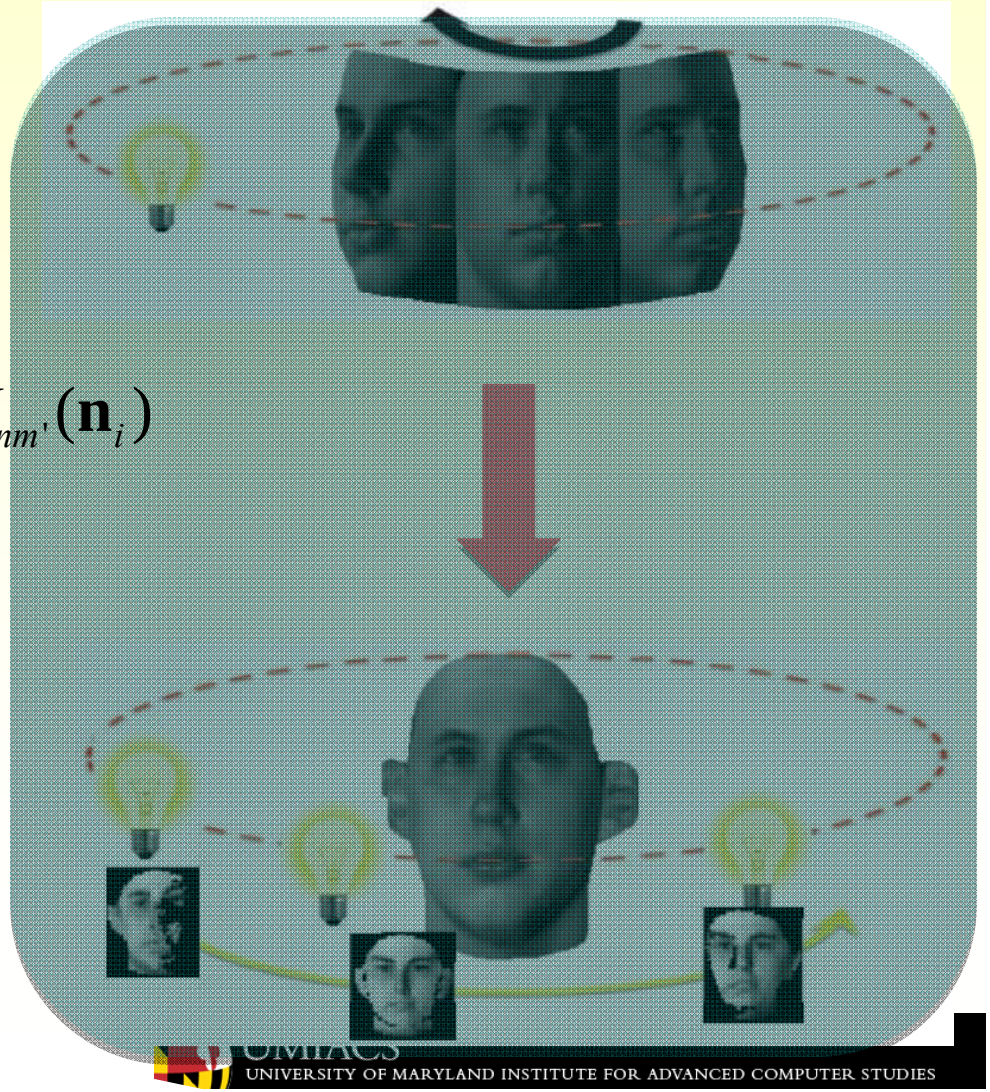
$$I_R(p_i) = \rho_i \sum_n \sum_m l_{nm} Y_{nm}(R(\mathbf{n}_i))$$

$$= \rho_i \sum_n \sum_m l_{nm} \sum_{m'} D^n_{mm'}(R) Y_{nm'}(\mathbf{n}_i)$$

$$= \rho_i \mathbf{L}^T (D(R)\mathbf{Y}(\mathbf{n}_i))$$

$$= \rho_i \mathbf{Y}(\mathbf{n}_i)^T (D(R)^T \mathbf{L})$$

Spherical harmonics transformation matrix



Albedo estimation using Kalman filter

$$P(\rho \mid \mathcal{Z}^t, \Theta) \propto P(Z_t \mid \rho, \mathcal{Z}^{t-1}, \Theta) P(\rho \mid \mathcal{Z}^{t-1}, \Theta)$$

Kalman observation equation:

$$P(Z_t \mid \rho, \mathcal{Z}^{t-1}, \Theta) \propto \mathcal{N}(I_t \mid H_t \rho, \Sigma_{v,t})$$

$$I_t = I_t(z_t, \theta_t) = \mathrm{diag}(\rho) H_t + \nu$$

$$H_t = diag(h_t = Y^T(n_i) L_t)$$

Kalman update equations:

$$\mu_{\rho,t} = \mu_{\rho,t-1} + K_t(I_t - H_t \mu_{\rho,t-1})$$

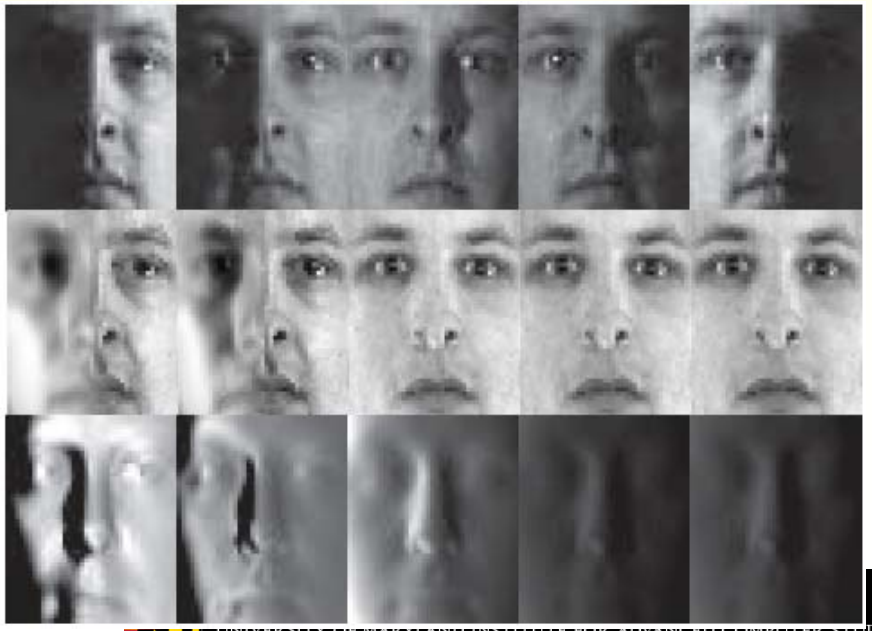
$$\Sigma_{\rho,t} = (eye - K_t)\Sigma_{\rho,t-1}$$

Kalman gain:

$$K_t = \Sigma_{\rho,t-1}(\Sigma_{\rho,t-1} + \Sigma_{v,t})^{-1}$$

Estimated albedo map

Kalman gain



Rao-Blackwellized Particle filter

- Since both pose and albedo are unknown → we have an augmented state space.

$$P(\rho, \theta_t \mid \mathcal{Z}^t, \Theta^{t-1})$$

- Particle filter needs a large set of particles to estimate the parameters of this augmented state space.
- In particle filtering, "Rao-Blackwellization" (RB) refers to integrating out a part of the state analytically, with the result that the variance of the resulting Rao-Blackwellized particle filter (RBPF) is sharply reduced.

RBPF leads to more accurate estimates of state parameters with fewer particles.

Pose tracking and albedo estimation

- Hybrid particle: $\{\theta_{t-1}^{(i)}, w_{t-1}^{(i)}, \mu_{\rho,t-1}^{(i)}, \Sigma_{\rho,t-1}^{(i)}\}$

- RBPF iterations over frames:

1. Sample from the dynamic model $P(\theta_t \mid \theta_{t-1}^{(i)})$ to obtain a predicted pose parameter $\bar{\theta}_t^{(j)}$.
2. Get the observation vector $I_t^{(j)}$
   - inverse warp of the 3D model on the current frame using $\bar{\theta}_t^{(j)}$ as the head pose and then find the intensity at the model vertices.
3. Update $\mu_{\rho,t-1}^{(i)}$ and $\Sigma_{\rho,t-1}^{(i)}$ according to the Kalman update equations
4. Calculate the importance weight $w_t^{(j)}$ as:

$$w_t^{(i)} \propto \exp(-\frac{1}{2}\|I_t^{(i)} - B_t^{(i)} L_t^{(i)}\|_{\Sigma_v}^2)$$

$$L_t^{(i)} = (B_t^{(i)})^{\dagger} I_t^{(i)}$$

Initialization

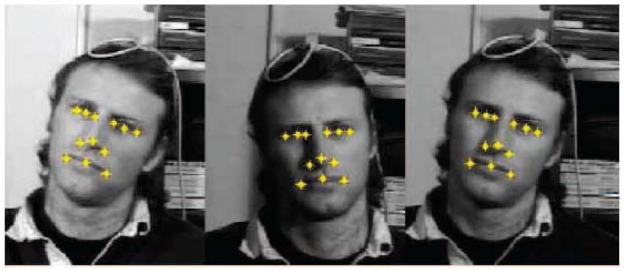
- 3D shape model is calculated using Vetter training data.
- 15 landmark points selected on the face (manually/automatically)
- Initial parameter estimation
  - θ_0..... using landmarks
  - $(\mu_{\rho,0}, \Sigma_{\rho,0})$.... using mean albedo and its variance over training data
  - $\Sigma_{v,0}$..... using training data

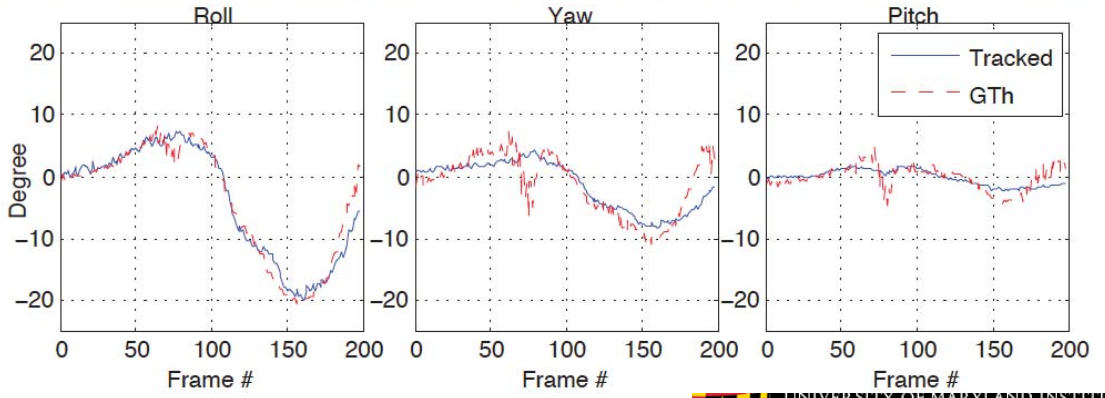
Pose tracking and albedo estimation

Illumination-invariant tracking

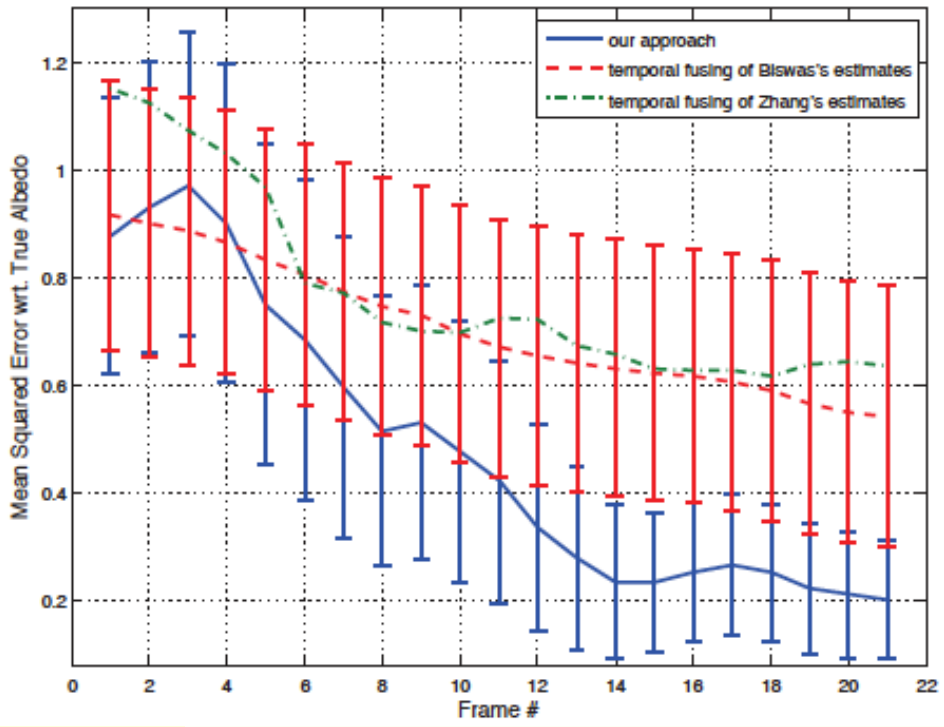
- The algorithm minimizes the projection error onto spherical harmonic subspace

$$w_t^{(i)} \propto \exp(-\frac{1}{2}\|I_t^{(i)} - B_t^{(i)} L_t^{(i)}\|_{\Sigma_v}^2)$$





Comparison



first frame last frame

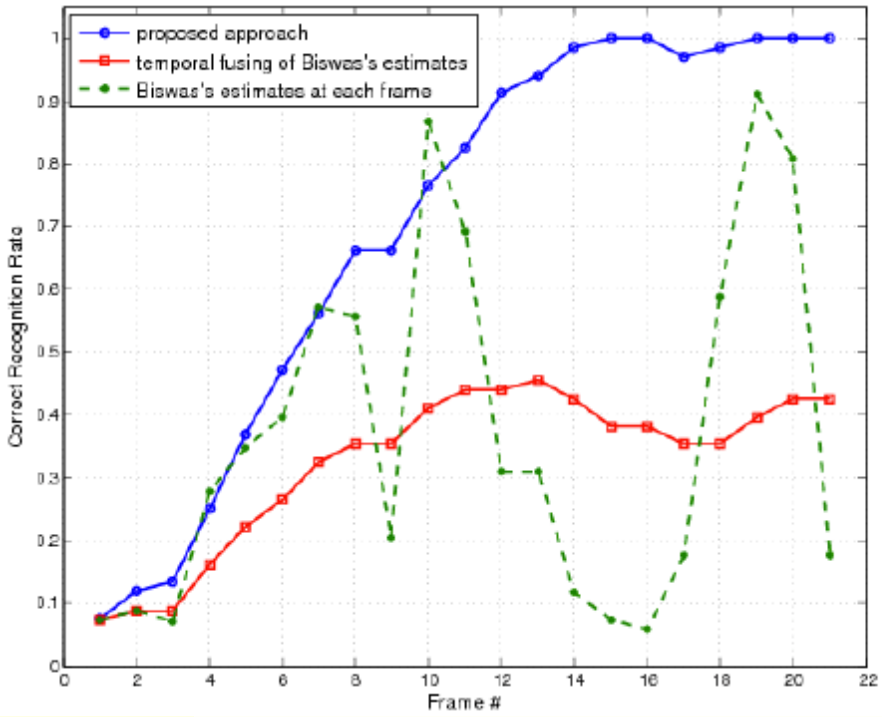
Our result

Temporal fusion Over Biswas's algorithm.

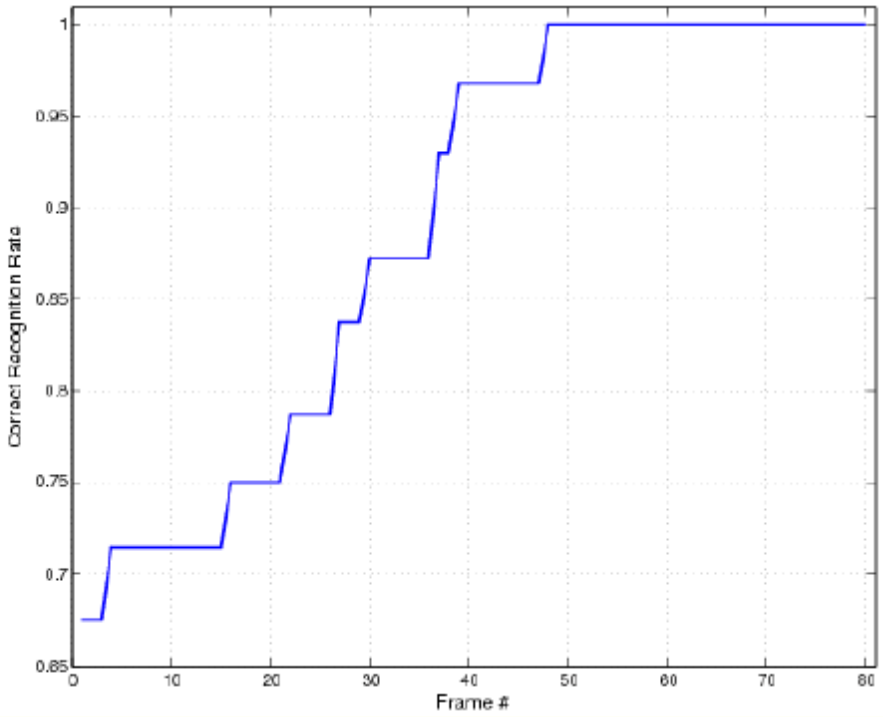
Temporal fusion Over Zhang's algorithm.

MSE of estimated albedo on synthetic PIE sequence (68 subjects)

Video-based face recognition



PIE dataset
(68 subjects, 21 images per subject)



BU dataset
6 subjects, 9 sequences per subject

Summary

- Discussed four methods for VFR
- Depends on tracker performance
- VFR methods that avoid tracking should be seriously considered
- Larger data sets are needed for evaluation
- Unconstrained video-based face recognition is a challenging problem.