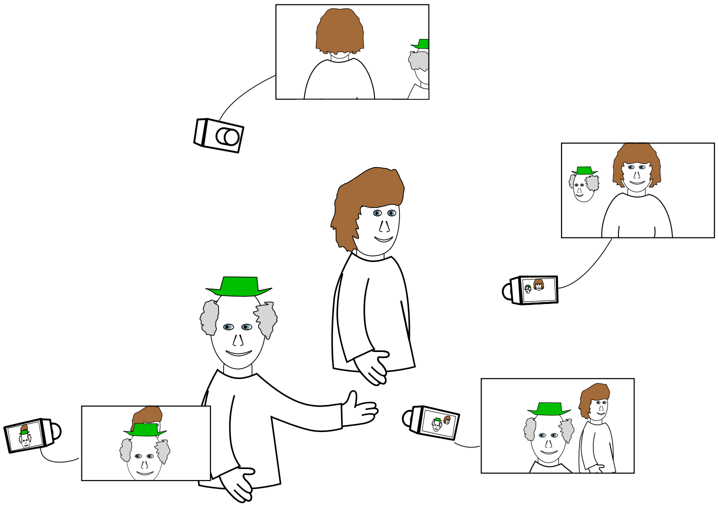
Cluster-Based Distributed Face Tracking in Camera Networks

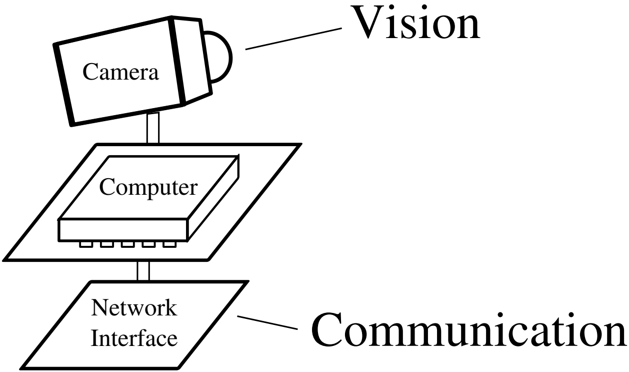
Josiah Yoder

Robot Vision Lab — Purdue University

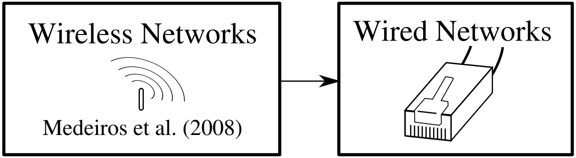
9 September 2011



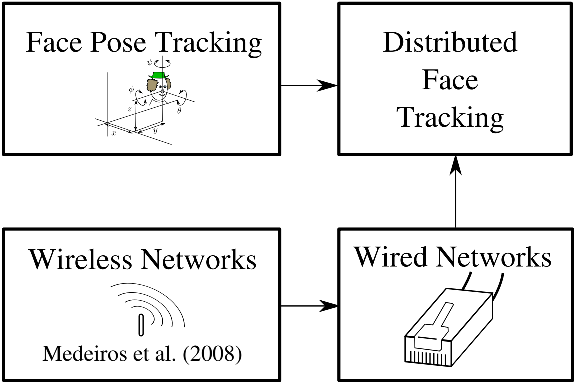
A Camera Node



OVERVIEW

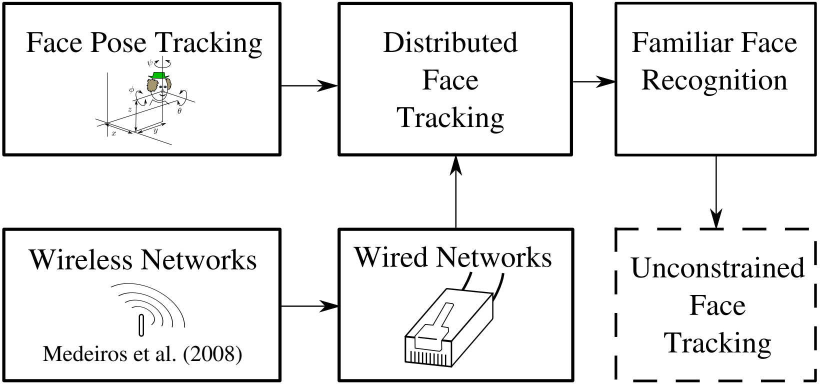


Overview

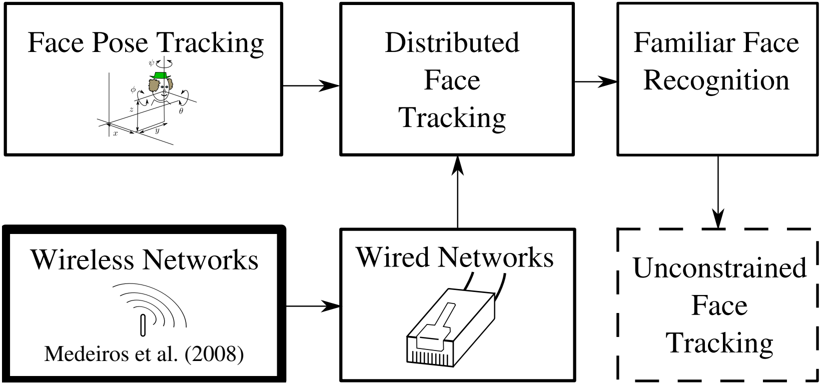


Face Pose Tracking
Distributed Face Tracking
Wireless Networks
Medeiros et al. (2008)
Wired Networks

Overview


Face Pose Tracking
Distributed Face Tracking
Familiar Face Recognition
Wireless Networks
Medeiros et al. (2008)
Wired Networks
Unconstrained Face Tracking

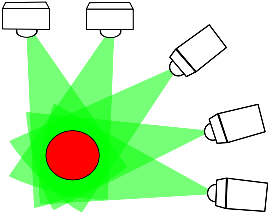
Cluster-based Tracking in Wireless Networks



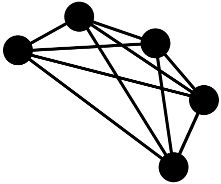
Cluster-based Tracking in Wireless Networks

- Developed by Medeiros et al. (2008)
- Addresses challenges of wireless networks:
  - Limited communication range
  - Cameras tracking same target may not be able to communicate

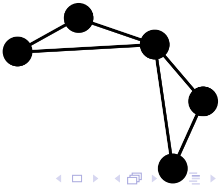
Vision Graphs & Communication Graphs



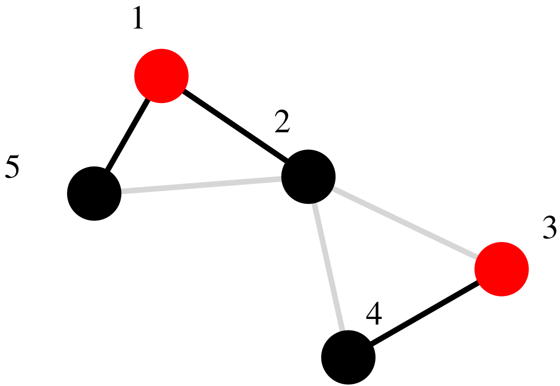
Vision Graph

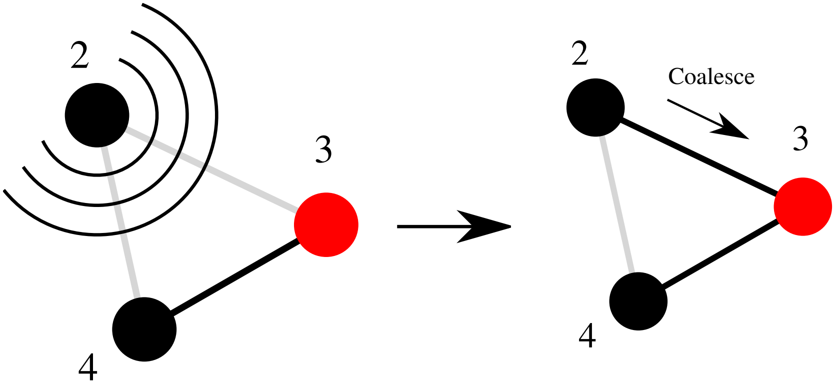


Communication Graph

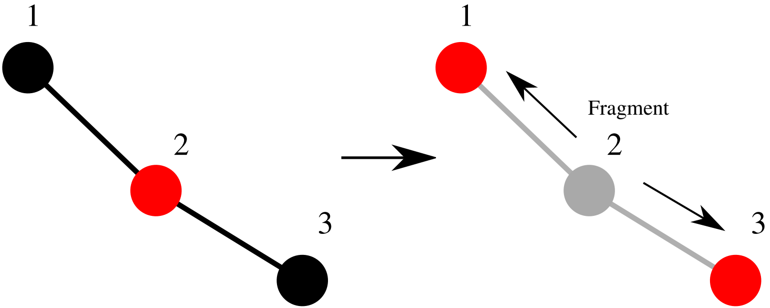


Cluster Leader Election

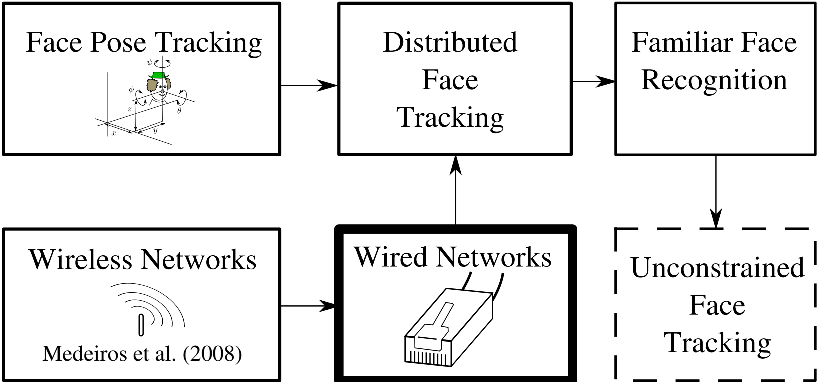



2
3
4
2
Coalesce
3
4

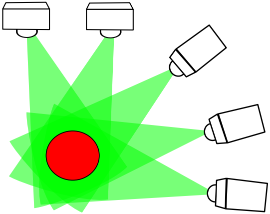
Cluster Fragmentation



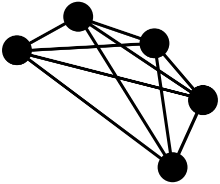
Tracking in Wired Networks



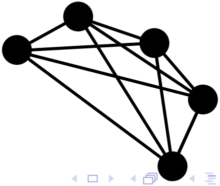
Tracking in Wired Networks



Vision Graph



Communication Graph

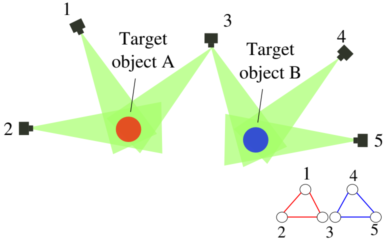


Establishing Corresponding Detections

- How can cameras determine they have detected the same target?
  - Detect objects
  - Extract visual features
  - Apply similarity criterion
  - Objects "match" if criterion passes a decision threshold
- Many variations in features and matching criteria
  - Color Histograms
  - HoG features
  - SIFT features
  - Point clouds
  - Face pose estimates
  - Gabor jets
  - ...

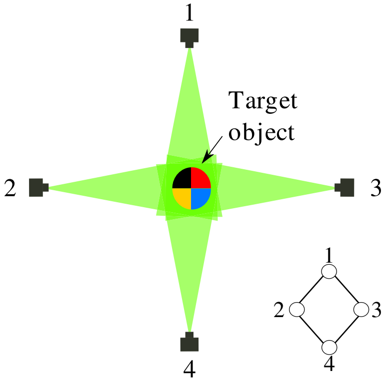
Tracking Graphs

- Edge between two cameras if their detected objects pass the matching criterion
- Cameras may participate in more than one tracking graph
- Dynamic: Changes as objects move



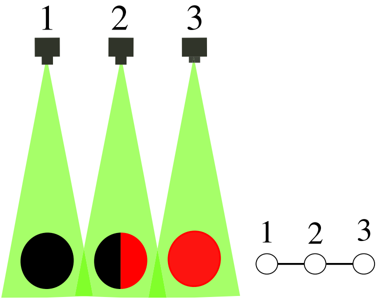
Tracking Graphs: Cluster Formation

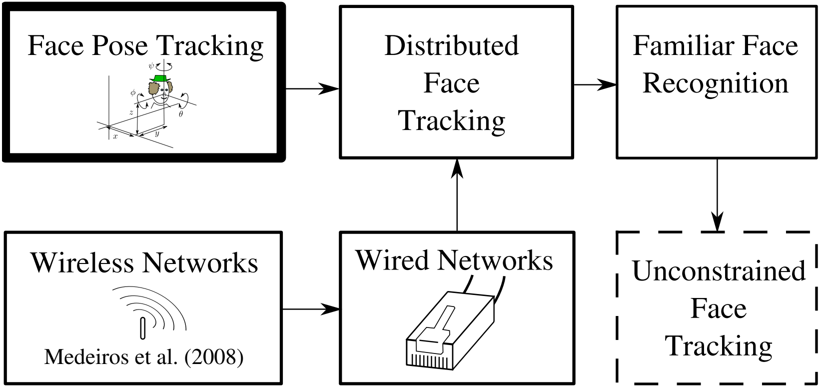
- May have missing edges in the tracking graph
- Cameras can only establish correspondence through other cameras
- For rapid cluster formation, cameras join clusters only with immediate neighbors



Tracking Graphs: Cluster Formation

- May have false edges in tracking graph
- Leads to a single cluster tracking multiple targets
- Fixed by cluster fragmentation during propagation



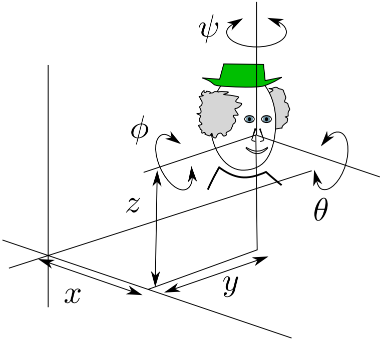

Face Pose Tracking
Distributed Face Tracking
Familiar Face Recognition
Wireless Networks
Medeiros et al. (2008)
Wired Networks
Unconstrained Face Tracking

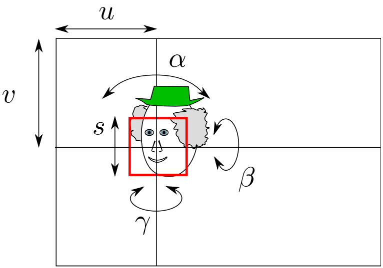
A Framework for Multi-Camera Face Tracking

A framework for face detection, pose estimation, and tracking:

- Allows generic single-camera methods to be incorporated into a multi-camera method
- Representation of face position as a coherent 6-DOF quantity

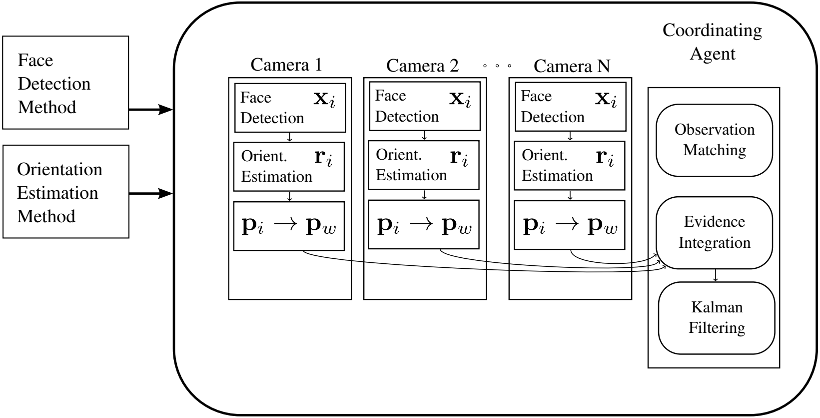
The 6-DOF World and Image-Based Poses





$$\mathbf{p}_w = [\underbrace{x, y, z,}_{\mathbf{x}_w^T,} \underbrace{\theta, \phi, \psi}_{\mathbf{r}_w^T}]^T = [\ \mathbf{x}_w^T, \quad \mathbf{r}_w^T\ ]^T$$

$$\mathbf{p}_i = [\underbrace{u, v, s,}_{\mathbf{x}_i^T,} \underbrace{\alpha, \beta, \gamma,}_{\mathbf{r}_i^T}]^T = [\ \mathbf{x}_i^T, \quad \mathbf{r}_i^T\ ]^T$$


Face Detection Method
Orientation Estimation Method
Camera 1
Camera 2
Camera N
Coordinating Agent
Face Detection $\mathbf{x}_i$
Orient. Estimation $\mathbf{r}_i$
$\mathbf{p}_i \rightarrow \mathbf{p}_w$
Observation Matching
Evidence Integration
Kalman Filtering

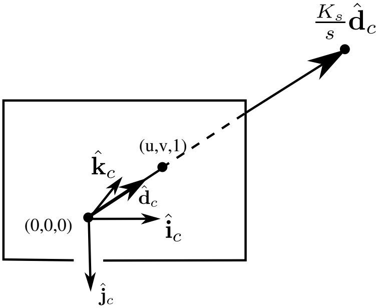
Transformation of Position and Rotation

- We can transform observations from image to world coordinates, through an invertible function $\mathbf{f}$

- $\mathbf{p}_w = \mathbf{f}(\mathbf{p}_i)$
- $\mathbf{p}_w = \begin{bmatrix} \mathbf{x}_w \\ \mathbf{r}_w \end{bmatrix} = \begin{bmatrix} \mathbf{f}_x(u,v,s) \\ \mathbf{f}_r(u,v,\alpha,\beta,\gamma) \end{bmatrix}$
- $\mathbf{x}_w = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad \mathbf{r}_w = \begin{bmatrix} \theta \\ \phi \\ \psi \end{bmatrix}$

- $\mathbf{f}_x(u,v,s)$ – [Iwaki et al. 2008]
- $\mathbf{f}_r(u,v,\alpha,\beta,\gamma)$ – [Murphy-Chutorian and Trivedi 2008]

Transformation of Position



$$\mathbf{x}_w = \mathbf{f}_x(\mathbf{x}_i) = {}_w\mathbf{R}_c \left(\frac{K_s}{s} \hat{\mathbf{d}}_c \right) + {}_w\mathbf{t}_c$$

$$\hat{\mathbf{d}}_c = (u\hat{\mathbf{i}}_c + v\hat{\mathbf{j}}_c + \hat{\mathbf{k}}_c)/(\sqrt{u^2 + v^2 + 1})$$

Transformation of Rotation

$$\mathbf{r}_w = \mathbf{f}_r(\mathbf{x}_i, \mathbf{r}_i) = [{}_w\mathbf{R}_c \, {}_c\mathbf{R}_i \, [\mathbf{r}_i]_{3\times 3}]_{3\times 1}$$

- ${}_w\mathbf{R}_c$ — Camera Rotation
- ${}_c\mathbf{R}_i$ — Murphy-Chutorian & Trivedi Rotation
- $[]_{3\times 3}$ — Conversion to Rotation Matrix
- $[]_{3\times 1}$ — Conversion to Roll-Pitch-Yaw

Murphy-Chutorian & Trivedi Rotation (2008)

rotation about the axis $\hat{\mathbf{k}}_c \times \hat{\mathbf{d}}_c$ by the angle $\cos^{-1}(\hat{\mathbf{k}}_c \cdot \hat{\mathbf{d}}_c)$

Uncertainty Modeling

We represent observations as Gaussian distributions

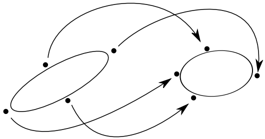
- Image-based observation:

$$\mathbf{p}_i \sim \mathcal{N}\left(\overline{\mathbf{p}}_i, \mathbf{C}_{\mathbf{p},i}\right)$$

- World observation:

$$\mathbf{p}_w \sim \mathcal{N}\left(\overline{\mathbf{p}}_w, \mathbf{C}_{\mathbf{p},w}\right)$$

- Transform from $\mathbf{p}_i$ to $\mathbf{p}_w$ using the Unscented Transform



Comparing Observations

We compute the distance between j^{th} and k^{th} observation using the Mahalanobis distance

$$d(\mathbf{p}_w^j, \mathbf{p}_w^k) = (\overline{\mathbf{p}}_w^j - \overline{\mathbf{p}}_w^k)^T (\mathbf{C}_{\mathbf{p},w}^j + \mathbf{C}_{\mathbf{p},w}^k)^{-1} (\overline{\mathbf{p}}_w^j - \overline{\mathbf{p}}_w^k)$$

Distributed:

- We declare two observations consistent if $d(\mathbf{p}_w^j, \mathbf{p}_w^k) < T$

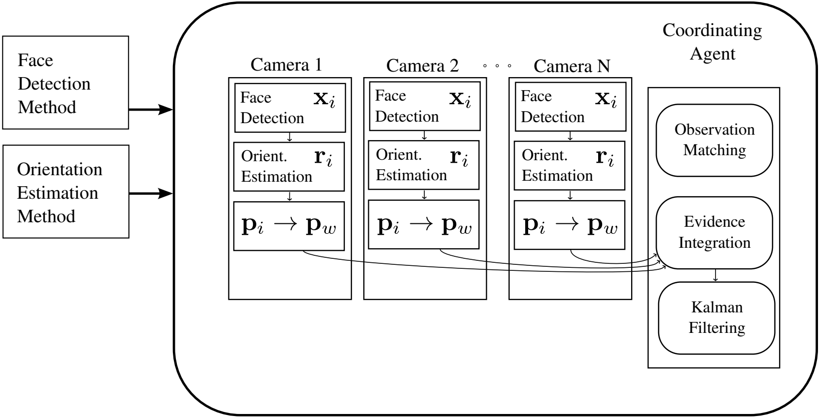
Centralized:

- We employ a feature clustering algorithm based on the distance $d(\mathbf{p}_w^j, \mathbf{p}_w^k) - T_{\text{clique}}$

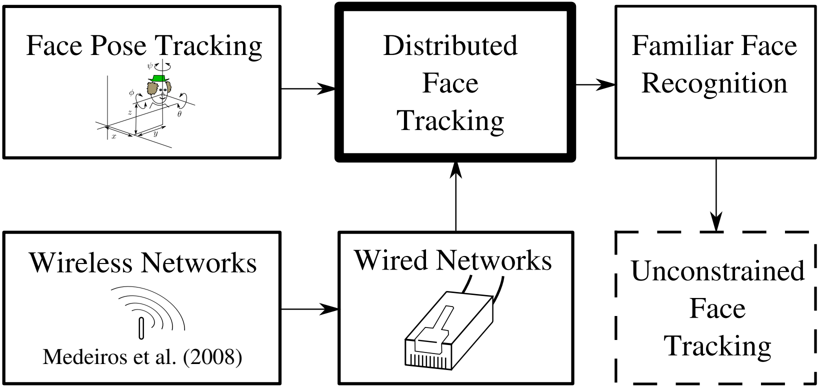
INTEGRATING OBSERVATIONS

We use a minimum-variance estimator to integrate world observations into a more accurate estimate

$$
\begin{aligned}
E\left[\hat{\mathbf{p}}_w\right] &= \left(Cov\left[\hat{\mathbf{p}}_w\right]\right) \sum_{\mathbf{p}_w^k \in \mathscr{E}} \left(\mathbf{C}_{\mathbf{p},w}^k\right)^{-1} \overline{\mathbf{p}}_w^k \\
Cov\left[\hat{\mathbf{p}}_w\right] &= \left(\sum_{\mathbf{p}_w^k \in \mathscr{E}} \left(\mathbf{C}_{\mathbf{p},w}^k\right)^{-1} \right)^{-1}
\end{aligned}
$$


Face
Detection
Method
Orientation
Estimation
Method
Camera 1
Camera 2
Camera N
Coordinating
Agent
Face
Detection
$\mathbf{x}_i$
Orient.
Estimation
$\mathbf{r}_i$
$\mathbf{p}_i \rightarrow \mathbf{p}_w$
Observation
Matching
Evidence
Integration
Kalman
Filtering

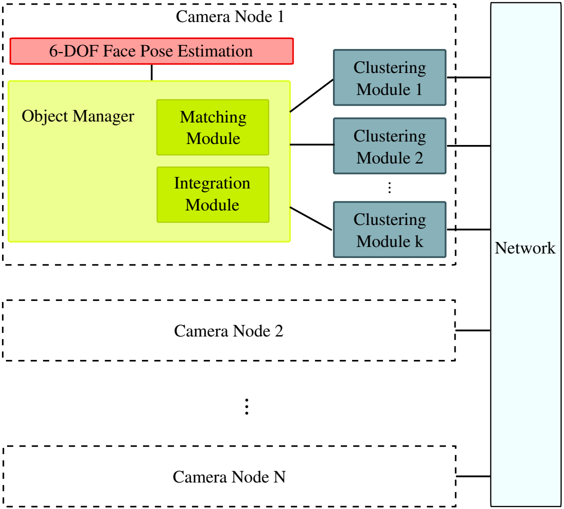
DISTRIBUTED CLUSTER-BASED FACE POSE TRACKING



Here we combine the two systems

- Multi-camera face pose tracking framework
- Cluster-based tracking protocol

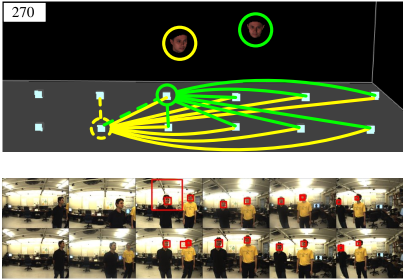
System Architecture

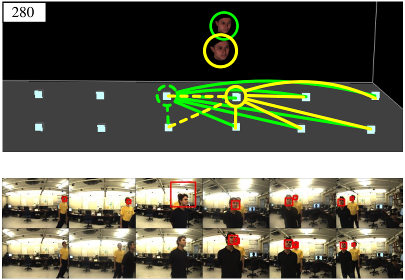


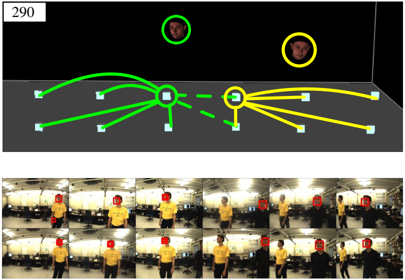
Face Pose as Identifying Feature

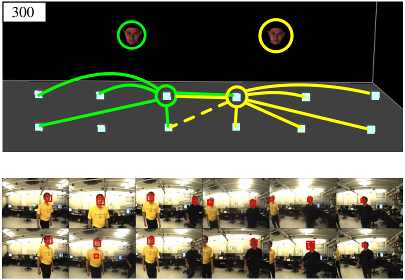
- Cluster-based protocol makes use of a feature to distinguish targets
- Real-time unconstrained face recognition not available
- We use current face pose for this feature

EXPERIMENTS

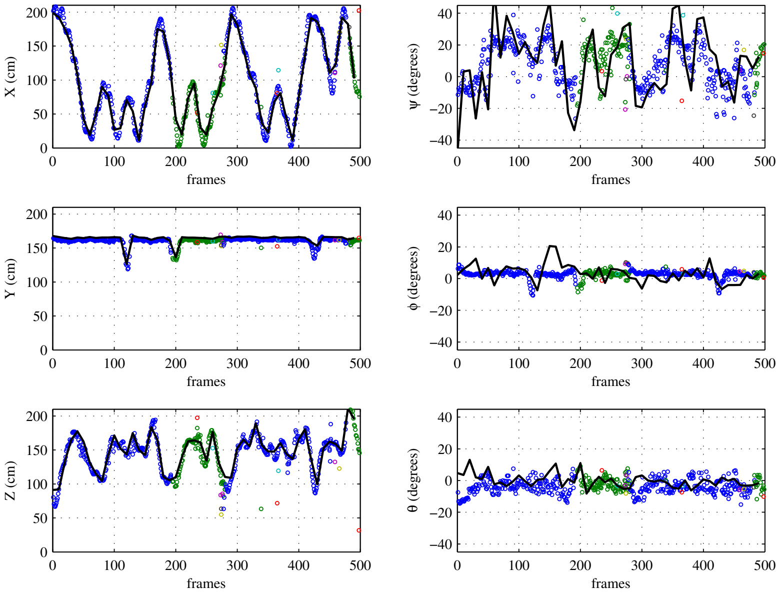








EXPERIMENTS



Experiments

Comparison with a centralized system

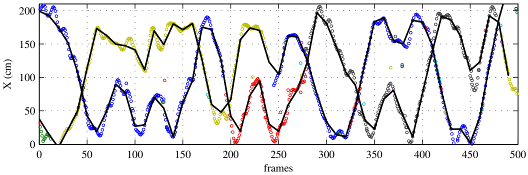
Both

- use same synchronization of collaboration period
- use 6-DOF face pose estimation framework
- detect faces in the individual cameras

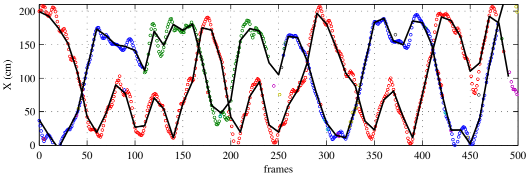
In centralized system

- 6-DOF observations are sent to a central server
- Correspondences are computed based on all pairwise matches

Distributed



Centralized



EXPERIMENTS

	TP	FP	$rmse_{\mathbf{T}}(cm)$	$rmse_{\mathbf{R}}(^{\circ})$
Centralized	95 (95%)	12 (12%)	5.8	20.8
Distributed	94 (94%)	4 (4%)	6.1	18.7

Face Recognition

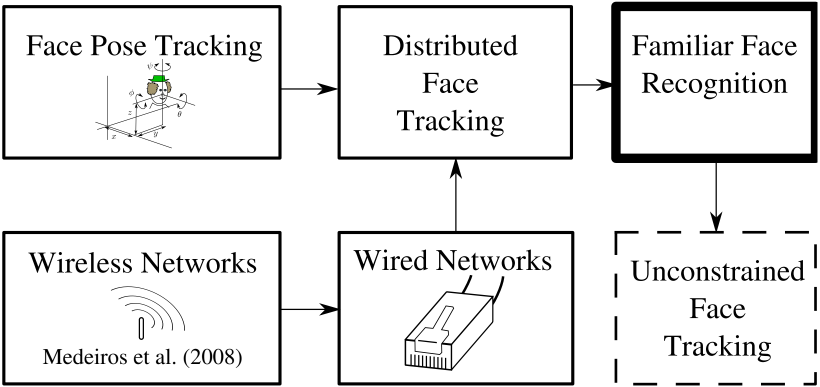
- Cluster-based protocol can also be used for other activities
- Here, we perform distributed face recognition
- Face recognition useful for multi-camera tracking
  - Associate observations from multiple cameras
  - Associate multiple tracks with the same person
  - Restore lost tracks
  - Many other applications
- Each camera performs PCA face recognition
  - Project face images into PCA space
  - Select nearest neighbor from training set (gallery) of faces
  - Send vote for that person to cluster leader
- Cluster leader counts votes and declares overall winner

Experiments: Distributed Face Recognition

Tracking TP / FP	Recognition TP / FP
92.4% / 7.6%	87.9% / 9.9%

- Completely distributed:
  - No central server, single point of failure
  - Scalable — Load on each link or node does not increase with network size
- Only using frontal faces . . . uncommon in camera networks

Human Familiarity-Benchmarked Face Recognition Database



Unconstrained Face Recognition

- Face recognition useful for multi-camera tracking
- Current algorithms poor for unconstrained face images
  - Low resolution
  - Varying pose
- Human performance is still significantly better for unconstrained images
- How to compare algorithms with the best human performance?

Benchmarks for Unconstrained Face Recognition

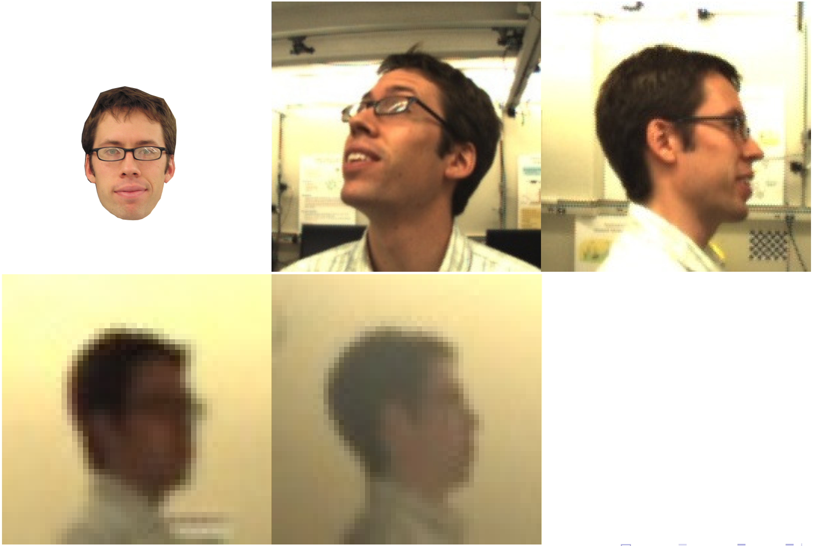
- The best human performance: "familiar" face recognition
- Unfair comparison?
  - People have prior knowledge unavailable to algorithms
  - Memories from previous encounters
  - Emotions, social relationships, etc.
- Can use same prior knowledge:
  - Videos of people to be pictured in the testing phase
- People can gain some sort of "familiarity" through watching the videos
- People can gain more familiarity if they chat while watching the videos (Bruce et al., 2001)

N-Viewer Familiarity Benchmark

- Familiarization: (training)
  - N people watch videos together
- Testing:
  - Each person performs the face matching recognition task separately
- Questions for familiarization:
  1. Which of your friends does he/she look like?
  2. What sports or hobbies might he/she like?
  3. What actor/actress or politician does he/she look like?
  4. What might his/her major be?
  5. Make a nickname for him/her.
  6. Describe his/her personality.
- Here, we use only 1-viewer familiarization.

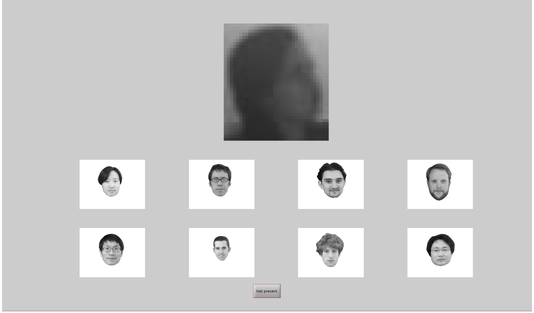
1-Viewer Benchmarked Database

DATABASE



1-Viewer Benchmarked Database

- 20 subjects tested for recognition matching ability
- Unfamiliar or 1-Viewer Familiar Test
- Familiar Test

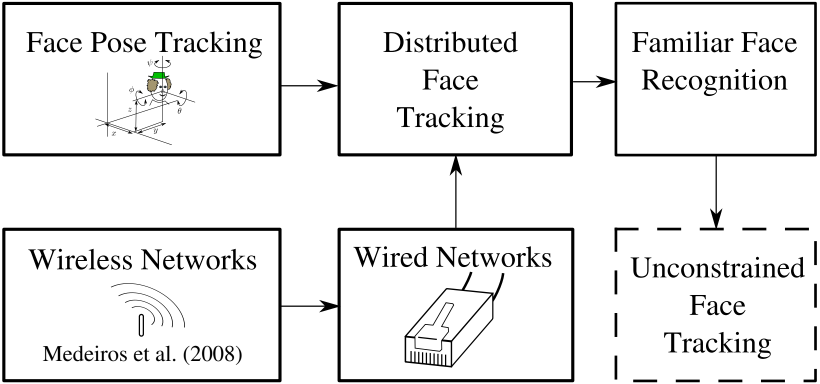


Benchmarked Performance

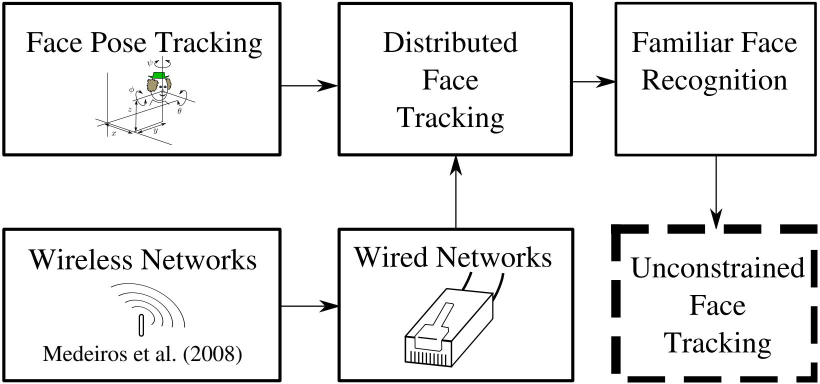
	% correct, mean (std)
Unfamiliar	54 (20)
1-Viewer Familiar	54 (17)
Previously Familiar	80 (19)

- 1-Viewer Familiarity does not improve performance
- Previous Familiarity does, even in challenging low-resolution images

CONCLUSIONS



Future Work


Face Pose Tracking
Distributed Face Tracking
Familiar Face Recognition
Wireless Networks
Medeiros et al. (2008)
Wired Networks
Unconstrained Face Tracking

Future Work

Cluster-based tracking in wired camera networks

- Coalescence and fragmentation without propagation

6-DOF face pose tracking

- Use local roll-pitch-yaw axes centered around current estimate or "bootstrapped" estimate

Familiar face recognition

- Unconstrained face pose tracking in camera networks

For Further Reading I

H. Iwaki, G. Srivastava, A. Kosaka, J. Park, and A. Kak.
A novel evidence accumulation framework for robust multi-camera person detection.
In Proceedings of the ACM/IEEE International Conference on Distributed Smart Cameras, pages 1–10, 2008.

H. Medeiros, J. Park, and A. Kak.
Distributed object trackng using a cluster-based Kalman filter in wireless camera networks.
In IEEE Journal of Selected Topics in Signal Processing, volume 2, pages 448–463, 2008.

For Further Reading II

E. Murphy-Chutorian and M. Trivedi.
3d tracking and dynamic analysis of human head movements and attentional targets.
Proceedings of the International Conference on Distributed Smart Cameras (ICDSC'08), 2008.

PRIOR WORK

- Face Detection, Pose Estimation, and Tracking
- Face Recognition
- Cluster-Based Tracking in Camera Networks

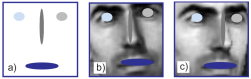
Prior Work: Face Pose Tracking

To track face pose, we need to:

- Detect faces
- Estimate face pose
- Track face pose

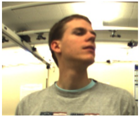
PRIOR WORK: FACE DETECTION

- Component-based detection

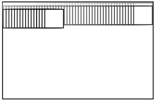


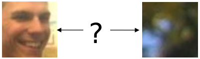
[Heisele et al. 2001]

- Color-based detection



- Scanning window methods

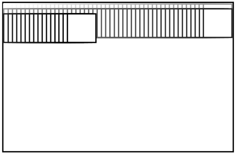




Heisele, B.; Serre, T.; Pontil, M. & Poggio, T. "Component-based face detection", Proc. CVPR, pp. 657–662. 2001.

Parameters Determined by Face Detection Methods

- Every face detection method determines
  - Face position
  - Face size
  - Face rotation (approximately)
- For example, for face size can be estimated:
  - from component distance
  - from color blob size
  - from window size



[Heisele et al. 2001]

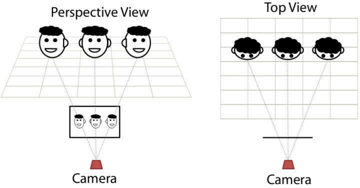
Prior Work: Single-Camera Pose Estimation

- There are many pose estimation methods as well [Survey: Murphy-Chutorian and Trivedi 2009]
  - Detector arrays
  - Regression methods
  - Deformable Models
  - etc.
- Methods estimate the face rotation
  - 1–, 2–, or 3–Degrees of Freedom (DOF)
  - Often analyze cropped face image and ignore the rest of the image
  - Rotation as if cropped image were at center of camera

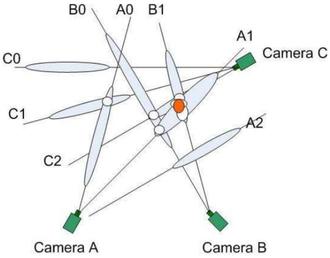
Prior Work: Single-Camera Face Tracking

- Pose tracking methods find a face iteratively based on the location in the previous frame
- Some of these methods analyze the cropped face image in image-based coordinates
    - Active Appearance Models
    - Appearance-template particle filters

Prior Work: Multi-camera Face Pose Estimation



[Murphy-Chutorian and Trivedi, 2008]

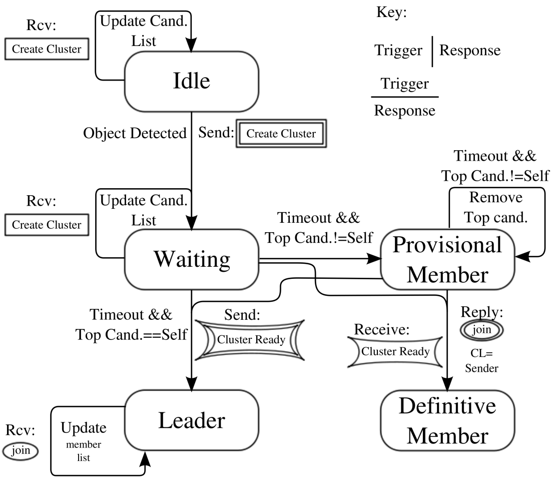


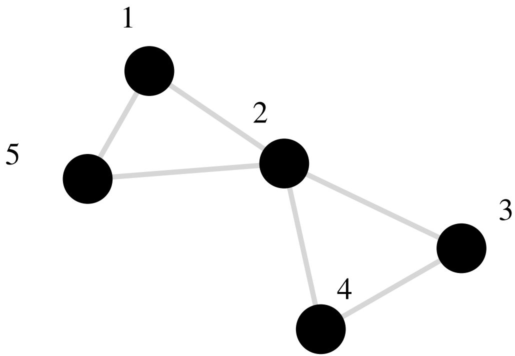
[Iwaki et al. 2008]

Prior Work: Cluster-based Tracking in Sensor Networks

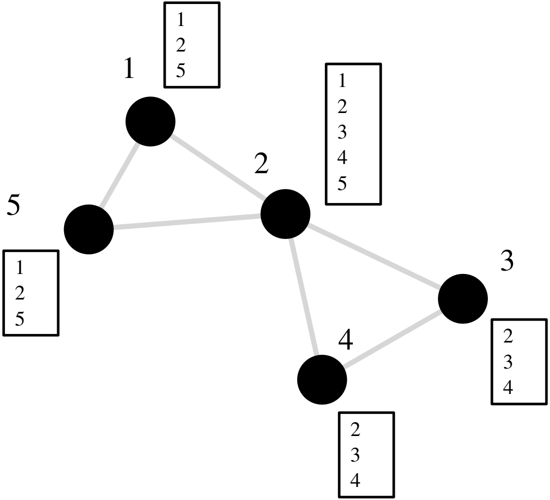
- General Sensor Networks:
  - Clusters used to facilitate data aggregation, e.g. for sensor monitoring
- Tracking:
  - A cluster is dedicated to tracking a single target
  - Cameras may participate in multiple clusters, track multiple targets
- Tracking protocols and systems:
  - Zhang and Cao (2004) organize clusters as Dynamic Convoy Trees
  - Blum et al. (2003) avoid creating multiple leaders tracking the same target using multi-hop communication
- No works prior to Medeiros et al. (2008) methods take into account the directional nature of camera sensors.

Cluster-based Communication Protocol: Cluster Leader Election

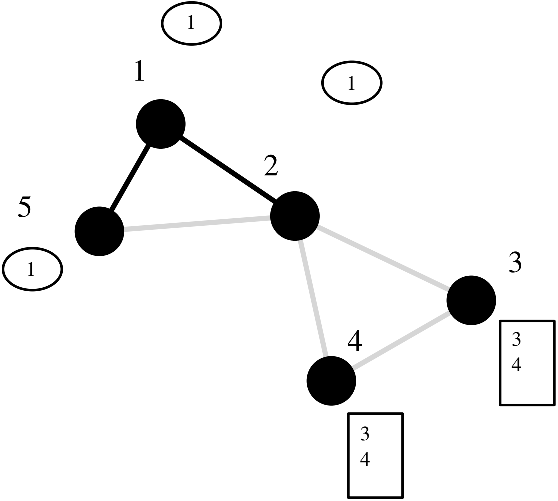



1
2
5
3
4

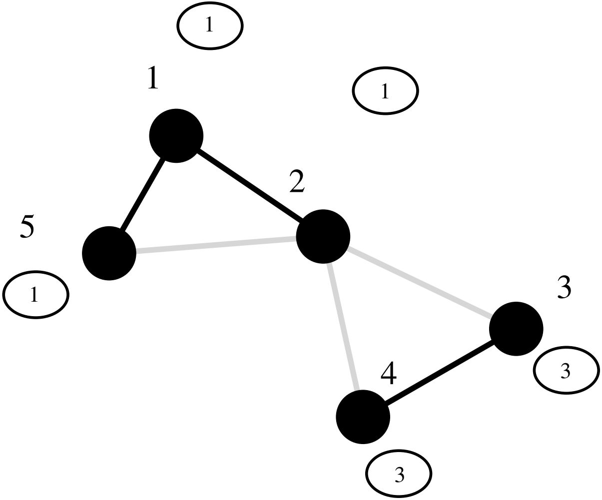
Cluster Leader Election



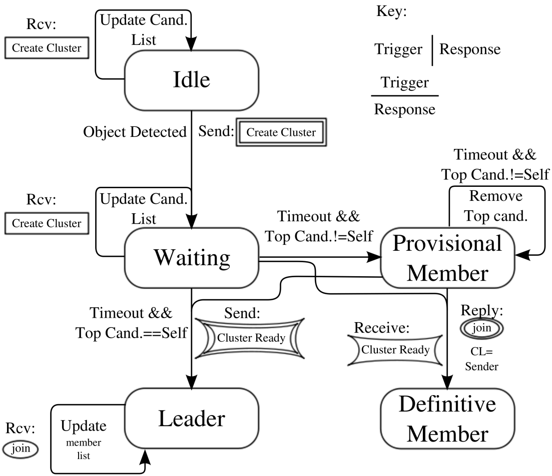
Cluster Leader Election



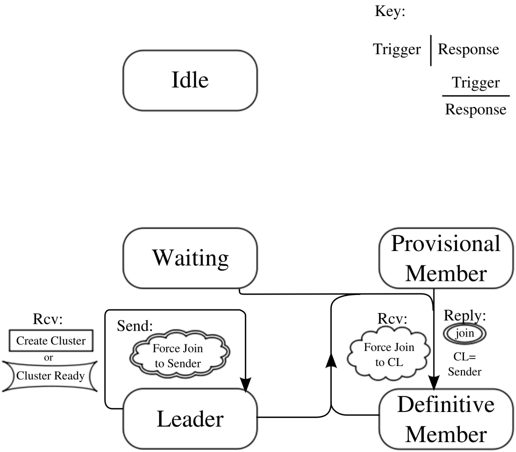
Cluster Leader Election



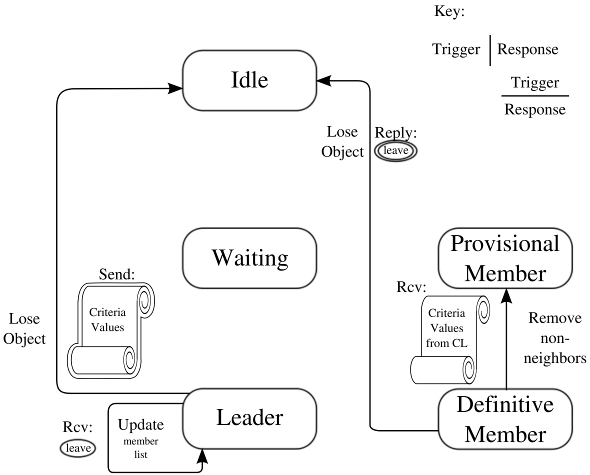
Cluster Leader Election



CLUSTER PROPAGATION (1)



Cluster Propagation (2)



Unscented Transformation

- We assume that $\mathbf{C}_{\mathbf{p},i}$ is diagonal
- We take a set of $2N$ sigma-points $\dot{\mathbf{p}}_i^k$ in image-based coordinates

$$\dot{\mathbf{p}}_i^k = \overline{\mathbf{p}}_i + \sqrt{N}\sigma_k \mathbf{e}_k \qquad k = 1,\ldots,N$$
$$\dot{\mathbf{p}}_i^{k+N} = \overline{\mathbf{p}}_i + \sqrt{N}\sigma_k \mathbf{e}_k \qquad k = 1,\ldots,N$$

Unscented Transformation

- Then transform each sigma point into world coordinates

$$\dot{\mathbf{p}}_w^k = \mathbf{f}(\dot{\mathbf{p}}_i^k)$$

- And compute the mean and covariance of the points in the world space

$$\overline{\mathbf{p}}_w = \frac{1}{2N} \sum_{k=1}^{2N} \dot{\mathbf{p}}_w^k$$

$$\mathbf{C}_{\mathbf{p},w} = \frac{1}{2N} \sum_{k=1}^{2N} \left(\dot{\mathbf{p}}_w^k - \overline{\mathbf{p}}_w \right) \left(\dot{\mathbf{p}}_w^k - \overline{\mathbf{p}}_w \right)^T$$