Computational methods and models for inference in multi-camera systems

Aswin C. Sankaranarayanan
Rice University
Broad research interests

Computer vision

Signal processing

Multi-camera systems
Tracking, video conferencing, evaluation, particle filtering

Complex reflectance

Compressive sensing
Video CS, union-of-subspace models

Image manifold models
Multi-camera systems are becoming increasingly popular

Surveillance

Entertainment

Video conferencing

Motion capture
There is a crisis ...

• Ever increasing number of cameras and sensors

• Deluge of data
  – Storage and processing costs are high
  – Wasteful
  – Poor scalability

• Large scope of applications
  – Different application domains have different challenges and concerns
Three research directions

• Efficient inference
  – Use geometry of imaging to enhance inference algorithms
  – Design robust statistical estimators for multi-camera fusion

• Reducing sensing costs
  – Novel camera designs along the lines of compressive sensing

• Concise representations
  – Manifold models for image ensembles
Efficient inference
Motivation

• Objects are imaged at different resolutions.

• As a consequence, accuracy of estimation of object location is NOT uniform over the plane and across multiple views.

How does camera positioning affect location estimation?
Imaging with a pinhole camera

A simple and accurate model of imaging is the **pinhole model**. Rays from the world pass through a pinhole and are imaged on a plane.

**Loss of depth** in the projection

This implies that given $X_L$, we can only say that the projection of $X$ onto “Right View” lies on a line. This constraint is called the **epipolar constraint**.

(Figures courtesy Wikipedia)
For **planar scenes**, we can uniquely map points across image planes.

This property is called the **homography**.

It is a **non-linear** one-to-one mapping from one image plane to another.
Solution outline

- Model the image plane location as a random variable (RV)
- Study how the distribution of the RV changes under the homography.
- The transformed RV’s statistics would decide the appropriate fusion scheme.
A toy problem: static localization

How to fuse these estimates?
Solution outline

• Model the image plane location as a random variable (RV)

• Study how the distribution of the RV changes under the homography.

• The transformed RV’s statistics would decide the appropriate fusion scheme.
Connections to the line at infinity

The distribution of the transformed random variable depends on the proximity of the image plane random variable to the projection of the line at infinity.

If the mean of the random variable $Z_u$ is close to the Line at Infinity, then $Z_x$ has a mixture density of which one part is Cauchy.

When the mean is far away from the Line at infinity, then the strength of the Cauchy distribution is negligible and we can approximate the transformed random variable using its moments.

[Proc. IEEE 2008]
• Densities on the world plane turn out to be approx. Normal if the Image plane densities are Normal too!

• Problem: How to estimate parameters of the world plane RVs
  – Linearization, Unscented Transformations.

• Example: Fusing Normal densities with Min. Variance estimators.
Extreme camera placement

Camera Views

Top View (metric rectified)

Log-Variance along x-direction

Log-variance along y-direction

White = High var.
Black = Low var.
Bound on localization error

Log-var. of \textbf{minimum variance estimator} along x-direction

Log-var. of \textbf{minimum variance estimator} along y-direction
Building a multi-camera tracking system

Challenges
• Data from varied sources
• Inter-camera Registration
• Multiple targets

Benefits of multi-camera fusion
• Ability to handle occlusion
• Accurate tracking.
Multi-camera tracking

6 cameras, 9 targets

Particle filter-based tracker

- data assoc using JPDAF
The FlexiView project
Bounds on localization error
General scenes

- Camera to camera mapping is no longer invertible
- Need to perform inference in 3D using **triangulation** as the key tool
Head tracking in multi-camera network
Pose tracking for video conferencing

• Knowledge of head pose is very useful in video conf.
  – **Augmented reality**, enhanced realism, compression, ....

• Challenges
  – Unknown camera placement (need to calibrate on the fly)
  – Real time constraints
Pose tracking results

• A real time system
  – 25 FPS tracking with <= 3 cameras
  – Extended tracking of pose due to multiple cameras
  – Keywords: Automatic initialization, self-calibration, 2D-3D correspondences, robust optimization, keyframes, ...

Joint work with MSR [AMFG2010]
Acoustic and video fusion

• Similar ideas extend to **acoustic arrays** as well
  – Direction of arrival of acoustic waves is a depth ambiguity much like the epipolar constraint

• Ability to maintain track even when one of the modalities fails

• Particle filter: Cloud indicates uncertainty in location.

[TMM 2008]
The data deluge problem

• More cameras == better inference

• More cameras == more data!
Parsimonious sensing
Conventional sensors

• The Shannon-Nyquist theorem
  – Reconstruct signals perfectly provided we sample at a rate twice at their bandwidth
  – At the heart of modern sensors and signal processing

• Yet, wasteful
  – Compression literature tells us that signals have additional structure
  – **Sparsity** of the signal in a transform basis → compressive sensing
Sparsity / compressibility

$N$ pixels

$N$ wideband signal samples

$K \ll N$

large wavelet coefficients

(blue = 0)

$K \ll N$

Large TF coefficients

$K \ll N$
Concise signal structure

- **Sparse** signal: only $K$ out of $N$ coordinates nonzero
  - model: union of $K$-dimensional subspaces aligned w/ coordinate axes
Compressive sensing

- **Sensing** via randomized dimensionality reduction

\[
M \times 1 \quad \text{Random or compressive measurements} \quad \Phi \quad N \times 1 \quad \text{sparse signal}
\]

- **Recovery:** solve an ill-posed inverse problem ... but exploit the geometrical structure of sparse signals

- When the measurement matrix satisfies certain properties, and \( M \) is large enough (\( O(K \ldots ) \) ) guarantees on recovery
CS and computer vision

• Sensor design
  – Novel cameras
  – Parsimonious sensing of light transport

• Underdetermined linear systems
  – Shape from shading
  – Super resolution

• Renewed interest in sparse approximation algorithms
  – Dictionary learning
  – Union-of-subspace models
  – Classification
Single-pixel camera

$\mathbf{x}$  $\mathbf{R}$  $\mathbf{y}$

$N = 65536$ pixels

$M = 11000$ measurements (16%)

$M = 1300$ measurements (2%)
From image to video sensing

• Not a simple extension of spatial acquisition techniques
  – Cant treat time as another spatial dimension.

• **Ephemerality** of temporal events
  – Transitory nature of time
  – Need to measure temporal events at their “information rate”
  – Fleeting events are hard to predict and capture
CSBS: Background subtraction from compressive measurements

• Model innovations over a static background

Reconstructed images

Difference between Row 2 and Row 1

Reconstructing innovations

Measurement rate

50%  5%  2%  1%  0.5%

[ECCV2008]
CSLDS: Video CS under LDS model

• Dynamic textures, sequence of LDS for activity analysis, etc...

• Applicability:
  – Frames of the video lie on a low dimensional subspace
  – Linearity applicable over short durations?
Linear dynamical system example

Video sequence

Few frames

Six key images

All frames can be estimated using linear combinations of SIX images
LDS model

Observation model
Data lies on a low dimensional subspace

C is a basis for the subspace. For uncompressed data, C is usually learnt using PCA on training examples.

\[ x_t = C \nu_t \]

\[ x_t \in \mathbb{R}^N, \nu_t \in \mathbb{R}^d, d \ll N \]

Terminology
\( x_t \) are the **frames** of the video (high dim)
\( \nu_t \) are the **subspace coefficients** (low dim)

xt are the frames of the video (high dim)
\( \nu_t \) are the subspace coefficients (low dim)
Random projection of subspace compressible data

• Assumption: Subspace compressibility
  – Rapid decay of eigenvectors of covariance of $y_t$

• Random projections $y_t = \Phi x_t, y_t \in \mathbb{R}^M$

$$[y_1, y_2, \ldots, y_T] = \Phi[x_1, x_2, \ldots, x_T] = \Phi C[v_1, v_2, \ldots, v_T]$$

$$\begin{pmatrix} y_1, y_2, \ldots, y_T \end{pmatrix} = \begin{pmatrix} \Phi \\ C \end{pmatrix} \begin{pmatrix} v_1, v_2, \ldots, v_T \end{pmatrix}$$

• SVD to recover the subspace coefficients?
Random Projection of Subspace Compressible Data

• How well does rows of $V^T$ approximate $[v_1, ..., v_T]$?
  – Inherent ambiguity in SVD

$$[y_1, y_2, ..., y_T] = \Phi[x_1, x_2, ..., x_T] = \Phi C[v_1, v_2, ..., v_T]$$

$$[y_1, y_2, ..., y_T] = USV^T$$

• **Lemma**: Suppose the singular values of $[x_1, ..., x_T]$ are $\lambda_1, \lambda_2, ..., \lambda_T$ then estimating $[v_1, ..., v_T]$ using $V^T$ leads to errors such that the $k$-th dimension has an error proportional to

$$\left( \lambda^2_{M+1} + ... + \lambda^2_T \right) / \lambda_k^2$$
Estimating “C”

• Suppose we take additional measurements

\[ \tilde{y}_t = \Phi_t x_t = \Phi_t C y_t \]

\[
\begin{bmatrix}
\tilde{y}_1 \\
\vdots \\
\tilde{y}_T
\end{bmatrix}
= 
\begin{bmatrix}
\Phi_1 C y_1 \\
\vdots \\
\Phi_T C y_T
\end{bmatrix}
\]

• We have an estimate of the subspace coefficients \([x_1, \ldots, x_T]\)
  
  – \( C \) is static!
  
  – Key idea: Accumulate measurements over time to solve for “C”
Overall algorithm

Scene

$X_{1:T}$

$Y_{1:T}$

Common measurements

Innovations measurements

$\Phi$

$\Phi_t$

$\tilde{Y}_{1:T}$

$\hat{v}_{1:T}$

Estimate state sequence using SVD

Model based CoSaMP for recovery of observation matrix

$\hat{X}_{1:T}$

$\hat{C}$

Key Points:
- Two sets of measurements at each time instant
  - Key step in handling bilinearity
- Measurement rate depends on subspace dimensionality “d” and sparsity level “K”
Reconstruction Results

Candle sequence
- 1024 frames (at 1000 fps)
- Resolution $64 \times 64$
- Measurement rate: 1.2 %
  - Comp = 81.92x
- State dimension: 15
- Reconstruction SNR: 13dB
- $M_{\text{common}} = 30$, $M_{\tilde{t}}=20$

LED sequence
- 500 frames (at 30 fps)
- Resolution $64 \times 64$
- Measurement rate: 2.1 %
  - Comp: 47.6x
- State dimension: 7
Comparisons

Original frames

Oracle LDS (25 dB)
CS-LDS (22 dB)

Frame-to-frame (11 dB)

Fire sequence [FALSE COLOR]
- 250 frames (at 30 fps)
- Resolution **128 x 128**
- Measurement rate: 0.42% (Comp. = 234x)
- M_common = 30, M_tilde = 40
- State dimension: 20

Estimated “C” Matrix: Ground truth (top) and estimates (bottom)
Classification results (in %) on the traffic databases for two different values of state space dimension $d$. Results are over a database of 254 videos, each of length 50 frames at a resolution of $64 \times 64$ pixels under a measurement rate of 4%.

<table>
<thead>
<tr>
<th></th>
<th>Expt 1</th>
<th>Expt 2</th>
<th>Expt 3</th>
<th>Expt 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle LDS</td>
<td>85.71</td>
<td>85.93</td>
<td>87.5</td>
<td>92.06</td>
</tr>
<tr>
<td>CS-LDS</td>
<td>84.12</td>
<td>87.5</td>
<td>89.06</td>
<td>85.71</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Expt 1</th>
<th>Expt 2</th>
<th>Expt 3</th>
<th>Expt 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle LDS</td>
<td>77.77</td>
<td>82.81</td>
<td>92.18</td>
<td>80.95</td>
</tr>
<tr>
<td>CS-LDS</td>
<td>85.71</td>
<td>73.43</td>
<td>78.1</td>
<td>76.1</td>
</tr>
</tbody>
</table>
Image representations
The pose manifold

• Natural instantiation for multi-view data

• Low dimensional nonlinear models

• Success of machine learning algorithms
  – LLE, ISOMAPs, Laplacian Eigenmaps, ...
Manifold learning

- $K=2$
- rotation and scale
Theory/practice disconnect:  

**Smoothness**

- Practical image manifolds are not smooth!
- If images have sharp edges, then manifold is everywhere non-differentiable [Donoho, Grimes]
- Image distances are meaningless unless when the manifold is finely samples

Local isometry

Local tangent approximations
Failure of tangent plane approx.

- Ex: **cross-fading** when synthesizing / interpolating images that should lie on manifold
Optical flow-based Transport

- **Idea:** Modeling the displacement field is much easier than modeling the image intensity field
  - Displacement field == optical flow for a large class of images

\[
I_2(x, y) = I_1(x + u(x, y), y + v(x, y))
\]
A new model for transport on image manifolds
Manifold learning

Data
196 images of two bears moving linearly and independently

Task
Find low-dimensional embedding
ML + Parameter estimation

Data
196 images of a cup moving on a plane

Task 1
Find low-dimensional embedding

Task 2
Parameter estimation for new images (tracing an “R”)
Summary

• Three key areas
  – Concise representations
  – Parsimonious sensing
  – Efficient Inference

• Richer interplay between these areas

• Fundamental ideas with application beyond multi-camera systems
Collaborators

Rama Chellappa (UMD)
Richard Baraniuk (Rice)

Volkan Cevher (EPFL)
Ming Du (UMD)
Marco Duarte (Duke)
Chinmay Hegde (Rice)
Sriram Nagaraj (Rice)
Dikpal Reddy (UMD)
Sima Taheri (UMD)
Pavan Turaga (UMD)

Zhengyou Zhang+team (Microsoft Res.)