Toward recovering 3D structure from a static image

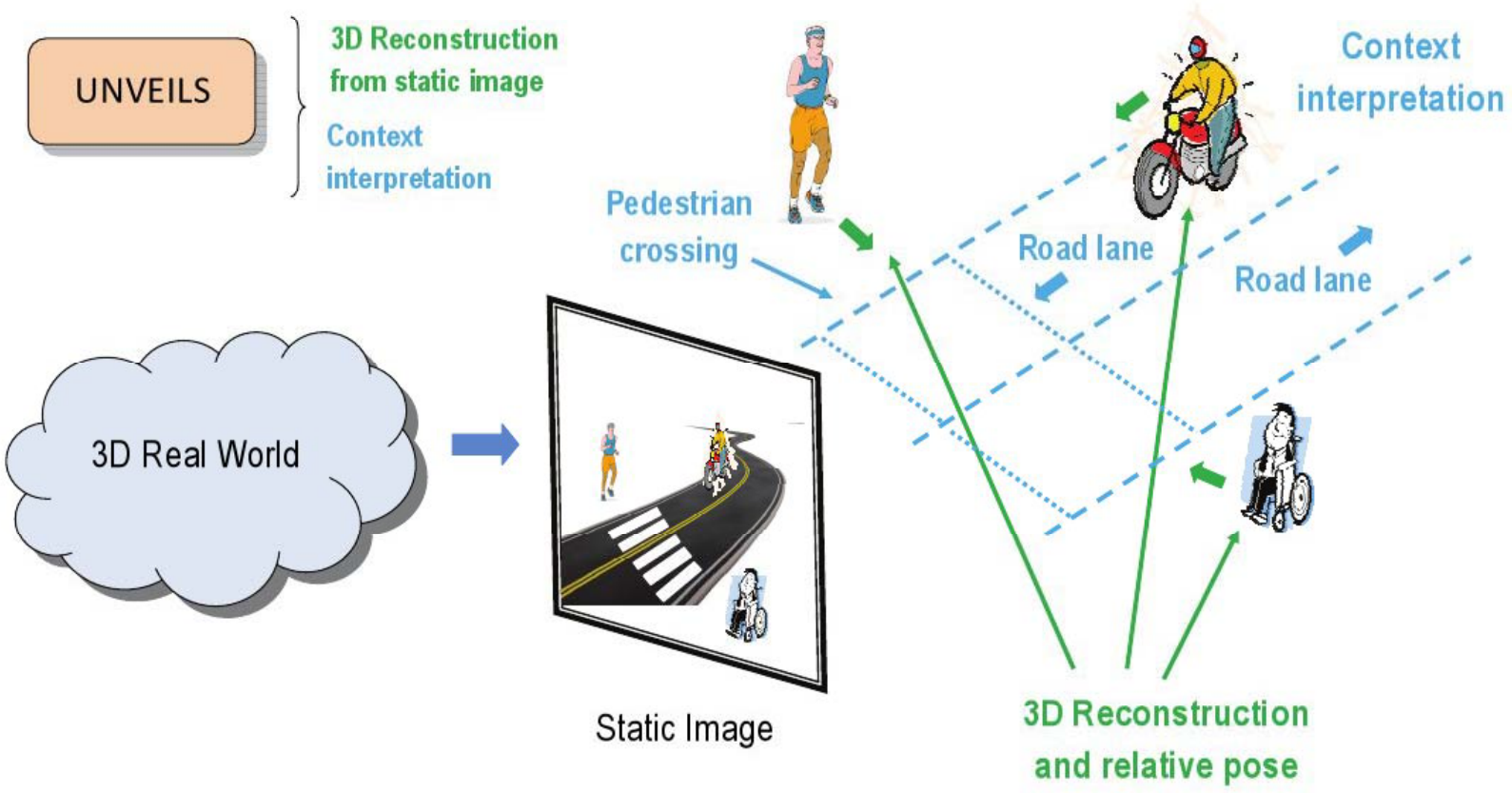
Motivation

- **Goal:**
  - To unveil the intrinsic 3D information contained in static images without using motion cues.
- **Evidence:**
  - Humans infer approximate 3D structure and pose from static images based on experience.
- **Proposal:**
  - To combine statistical generative models and parts-based object recognition techniques.
  - To apply geometric and anthropometric constraints for lifting the feasible poses of objects into 3D.
  - To create a massive dataset containing 3D-2D data of objects (time to 3D scan the world).

Potential Applications

- **ITS:**
  - Prediction of pedestrians intentions.
  - Traffic Monitoring.
  - Scene understanding.
- **Robotics:**
  - 3D object recognition.
  - Human-machine interaction.
- **Others:**
  - Sports and rehabilitation medicine, motion pictures, human recognition through gait analysis, security and surveillance, markerless motion capture, and gaming.

UNVEILS Vision



UNVEILS – UNwilding Visually Embedded Information in Latent Spaces

Steps

1. **Dataset design and creation (3D & 2D).**
2. Probabilistic learning of 3D pose.
3. Lifting the 3D pose from a static image.
4. Linkage between 2D-based object recognition and 3D-based pose learning.

Dataset creation

- Target:
  - 12 classes of most common objects (person, cyclist, dog, car, motorbike, truck, bus, chair, table, bottle, sofa, telephone).
  - >1 Million samples per class.
  - Consider different sub-behaviours and sub-classes.
  - Re-use of already existing datasets.

- Capturing object data (3D-2D):
  - Articulated objects: motion capture systems.
  - Non-articulated objects:
    - Small: Kinect.
    - Large: Velodyne.

Dataset creation

- Articulated objects:
  - Re-use of already existing datasets.
    - HumanEVA.
    - Human3.6M.
  - Enhancement of models for persons.
    - Different behaviours (walking, running, jumping, dancing, gesticulating, fighting, crouching down, sitting, etc).
    - Different users.
    - Different dynamics.
  - Objects of cyclists and dogs will be created from scratch.

Dataset creation

- Non-Articulated objects:
  - Re-use of already existing datasets.
    - University of Stuttgart.
  - Massive creation of new models.
    - Sensors:
      - Small object: Kinect.
      - Large objects: Velodyne.
    - Technique:
      - Pair-wise registration of consecutive images.
      - Correction of correlation matrices in horizontal and vertical directions.

Dataset creation

- Process:
  - Pair-wise registration of consecutive frames
    - Features detection in consecutive images, at t and t-1, using SURF
    - Matching filter (symmetry test)
    - Apply RANSAC to the point clouds at t and t-1
    - Inliers are used to compute the rotation and translation matrices using the quaternion-based algorithm
    - Compute mass centre after pair-wise registration (useful for further error correction)

Dataset creation

- Process:
  - Global registration
    - Global rotation matrix is computed and total rotation error is estimated
    - Rotational error distribution is computed and applied to rotation error compensation

$$\hat{\mathbf{R}}_{k,k+1} = \mathbf{E}_{k-1,k}^{<1/n>}\mathbf{R}_{k,k+1} = \mathbf{R}_{k,k+1}\mathbf{E}_{k,k+1}^{<1/n>}$$

    - Translation is decoupled from rotation

$$\tilde{\mathbf{t}}_{1,2} = (\mathbf{R}_{1,2} - \hat{\mathbf{R}}_{1,2})\mathbf{c}_2 + \mathbf{t}_{1,2}$$

    - The accumulated translation error is computed and distributed among the different steps using the decoupled rotation matrices

$$\hat{\mathbf{R}}_{k,k+1}\cdots\hat{\mathbf{R}}_{k-2,k-1}\hat{\mathbf{t}}_{k-1,k} + \cdots + \hat{\mathbf{R}}_{k,k+1}\hat{\mathbf{t}}_{k+1,k+2} + \hat{\mathbf{t}}_{k,k+1} = 0$$

Dataset creation

- Example:
  - Process
  - Result

Steps

1. Dataset design and creation (3D & 2D).
2. **Probabilistic learning of 3D pose.**
3. Lifting the 3D pose from a static image.
4. Linkage between 2D-based object recognition and 3D-based pose learning.

Probabilistic Learning of 3D Pose

- The goal is to perform dimensionality reduction in a probabilistic framework.

3D pose -> latent variable

$$\mathbf{y}_n = \mathbf{W}\mathbf{x}_n + \eta_n$$

$Y = [y_1 \ldots y_N]^T$ centred D-dimensional data

x_n q-dimensional latent variable

$W \in \Re^{Dxq}$

$\eta_n \in \Re^{Dx1}$

$$p(\eta_n) = N(\eta_n | \mathbf{0}, \beta^{-1}\mathbf{I})$$

Probabilistic Learning of 3D Pose

- Likelihood for a data point

$$p(\mathbf{y}_n|\mathbf{x}_n, \mathbf{W}, \beta) = N(\mathbf{y}_n|\mathbf{W}\mathbf{x}_n, \beta^{-1}\mathbf{I})$$

- Marginal likelihood (integration over latent variables)

$$p(\mathbf{y}_n|\mathbf{W}, \beta) = \int p(\mathbf{y}_n|\mathbf{x}_n, \mathbf{W}, \beta)\, p(\mathbf{x}_n)\, d\mathbf{x}_n$$

- Prior distribution over x_n

$$p(\mathbf{x}_n) = N(\mathbf{x}_n|\mathbf{0}, \mathbf{I})$$

Probabilistic Learning of 3D Pose

- Marginal Likelihood

$$p(\mathbf{y}_n|\mathbf{W},\beta) = N(\mathbf{y}_n|\mathbf{0},\mathbf{W}\mathbf{W}^{\mathrm{T}} + \beta^{-1}\mathbf{I})$$

- Likelihood of full data set (independent points)

$$p(\mathbf{Y}|\mathbf{W},\beta) = \prod_{n=1}^{N} p(\mathbf{y}_n|\mathbf{W},\beta)$$

- Parameters W can be found through maximisation using Probabilistic PCA (e.g. PCA + EM)

Probabilistic Learning of 3D Pose

- PPCA through Latent Variable Optimisation: marginalise parameters W and maximise variables X

$$p(\mathbf{W}) = \prod_{i=1}^{D} N(\mathbf{w}_i|\mathbf{0},\mathbf{I})$$

- Marginalised Likelihood

$$p(\mathbf{Y}|\mathbf{X},\beta) = \prod_{d=1}^{D} p(\mathbf{y}_{:,d}|\mathbf{X},\beta)$$

- where

$$p(\mathbf{y}_{:,d}|\mathbf{X},\beta) = N\left(\mathbf{y}_{:,d}|\mathbf{0},\mathbf{X}\mathbf{X}^{\mathrm{T}} + \beta^{-1}\mathbf{I}\right)$$

Probabilistic Learning of 3D Pose

- Optimisation of latent variables (minimisation of negative log likelihood)

$$L = -\frac{DN}{2}\ln 2\pi - \frac{D}{2}\ln|\mathbf{K}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\right)$$

- where

$$\mathbf{K} = \mathbf{X}\mathbf{X}^{\mathrm{T}} + \beta^{-1}\mathbf{I}$$

- Gaussian Process Latent Variable Model (GPLVM): the kernel becomes a Gaussian function

$$k(\mathbf{x}_i, \mathbf{x}_j) = \theta_{\mathrm{rbf}}\exp\left(-\frac{\gamma}{2}(\mathbf{x}_i - \mathbf{x}_j)^{\mathrm{T}}(\mathbf{x}_i - \mathbf{x}_j)\right) + \theta_{\mathrm{bias}} + \theta_{\mathrm{white}}\delta_{ij}$$

Probabilistic Learning of 3D Pose

- Gaussian Process Dynamical Model (GPDM): latent variable dynamical model
  - Low dimensional latent space
  - Probabilistic mapping from the latent space to the pose space
  - Dynamical model in the latent space (Φ_i and ψ_j are non-linear functions)

$$\mathbf{x}_t = \sum_i \mathbf{a}_i \, \phi_i(\mathbf{x}_{t-1}) + \mathbf{n}_{x,t}$$

$$\mathbf{y}_t = \sum_j \mathbf{b}_j \, \psi_j(\mathbf{x}_t) + \mathbf{n}_{y,t}$$

- where

$$\mathbf{y}_t \in \mathcal{R}^D$$

$$\mathbf{x}_t \in \mathcal{R}^d$$

Probabilistic Learning of 3D Pose

- Multivariate Gaussian data likelihood after marginalising parameters B (Scaled GPLVM)

$$p(\mathbf{Y} \mid \mathbf{X}, \bar{\beta}) = \frac{|\mathbf{W}|^N}{\sqrt{(2\pi)^{ND}|\mathbf{K}_Y|^D}} \exp\left(-\frac{1}{2}\mathrm{tr}\left(\mathbf{K}_Y^{-1}\mathbf{Y}\mathbf{W}^2\mathbf{Y}^T\right)\right)$$

- where

$$k_Y(\mathbf{x}, \mathbf{x}') = \beta_1 \exp\left(-\frac{\beta_2}{2}||\mathbf{x} - \mathbf{x}'||^2\right) + \frac{\delta_{\mathbf{x},\mathbf{x}'}}{\beta_3}$$

$$\mathbf{W} \equiv \mathrm{diag}(w_1, ..., w_D)$$

Probabilistic Learning of 3D Pose

- Density over latent trajectories after marginalising parameters A

$$p(\mathbf{X} \mid \bar{\alpha}) = \frac{p(\mathbf{x}_1)}{\sqrt{(2\pi)^{(N-1)d}|\mathbf{K}_X|^d}} \exp\left(-\frac{1}{2}\mathrm{tr}\left(\mathbf{K}_X^{-1}\mathbf{X}_{out}\mathbf{X}_{out}^T\right)\right)$$

- where

$$k_X(\mathbf{x}, \mathbf{x}') = \alpha_1 \exp\left(\frac{-\alpha_2}{2}||\mathbf{x} - \mathbf{x}'||^2\right) + \alpha_3 \mathbf{x}^T\mathbf{x}' + \frac{\delta_{\mathbf{x},\mathbf{x}'}}{\alpha_4}$$

$$\mathbf{X}_{out} = [\mathbf{x}_2, ..., \mathbf{x}_N]^T$$

- K_x is the (N-1)x(N-1) kernel matrix constructed from

$$\mathbf{X}_{in} = [\mathbf{x}_1, ..., \mathbf{x}_{N-1}]$$

Probabilistic Learning of 3D Pose

- Learning Process: estimation of latent positions and kernel hyperparameters. GPDM posterior

$$p(\mathbf{X}, \bar{\alpha}, \bar{\beta} \,|\, \mathbf{Y}) \;\propto\; p(\mathbf{Y} \,|\, \mathbf{X}, \bar{\beta})\, p(\mathbf{X} \,|\, \bar{\alpha})\, p(\bar{\alpha})\, p(\bar{\beta})$$

- Prior distributions over parameters

$$p(\bar{\alpha}) \;\propto\; \textstyle\prod_i \alpha_i^{-1}, \text{ and } \; p(\bar{\beta}) \;\propto\; \prod_i \beta_i^{-1}$$

- Minimisation of the negative log posterior

$$\begin{aligned}\mathcal{L} \;=\; & \frac{d}{2} \ln |\mathbf{K}_X| \;+\; \frac{1}{2} \mathrm{tr}\left(\mathbf{K}_X^{-1} \mathbf{X}_{out} \mathbf{X}_{out}^T\right) \\ & - N \ln |\mathbf{W}| \;+\; \frac{D}{2} \ln |\mathbf{K}_Y| \;+\; \frac{1}{2} \mathrm{tr}\left(\mathbf{K}_Y^{-1} \mathbf{Y} \mathbf{W}^2 \mathbf{Y}^T\right) \\ & + \sum_i \ln \alpha_i \;+\; \sum_i \ln \beta_i \;+\; C\end{aligned}$$

Probabilistic Learning of 3D Pose

- Example:

Video from HumanEVA dataset

Probabilistic Learning of 3D Pose

- Example using SGPLVM (no dynamics)
  - Out 1
  - Out 2

Probabilistic Learning of 3D Pose

- Example using GPDM (with dynamics)
  - Out 3

Steps

1. Dataset design and creation (3D & 2D).

2. Probabilistic learning of 3D pose.

3. **Lifting the 3D pose from a static image.**

4. Linkage between 2D-based object recognition and 3D-based pose learning.

Lifting the 3D Pose from a static image

- Posterior distribution to maximise

$$P(\phi_t|I_t,M) \propto P(I_t|\phi_t)\cdot P(\phi_t|M) = P(I_t|\phi_t)\cdot P(x_t,y_t|M)$$

with I_t (image), $\Phi_t=x_t,y_t,g_t$ (global position), M (training data)

- where

$$P(I_t|\phi_t) = \prod_{j=1}^{J} e^{-\frac{\left\|\hat{m}_t^j - P(p^j(\phi_t))\right\|^2}{2\sigma_e^2}}$$

$$P(x_t,y_t|M) = \frac{1}{\sqrt{\sigma_{x_t}^{2D}}} e^{-\frac{\left\|W(y_t-\mu_y(x_t))\right\|^2}{2\sigma^2(x_t)}} \cdot e^{-\frac{\|x_t\|^2}{2}}$$

Lifting the 3D Pose from a static image

- with

$$\mu_y(x_t) = \mu + Y^T K_Y^{-1} K_Y(x_t)$$

$$\sigma^2(x_t) = k_y(x_t, x_t) - k_y(x_t)^T K_Y^{-1} k_y(x_t)$$

- Final negative log likelihood to minimise

$$-lnP(\phi_t|I_t, M) = \frac{1}{2\sigma_e^2}\sum_{j=1}^{J}\left\|\hat{m}_t^j - P\left(p^j(\phi_t)\right)\right\|^2 + \frac{\left\|W(y_t - \mu_y(x_t))\right\|^2}{2\sigma^2(x_t)} + \frac{D}{2}ln\sigma^2(x_t) + \frac{1}{2}\|x_t\|^2$$

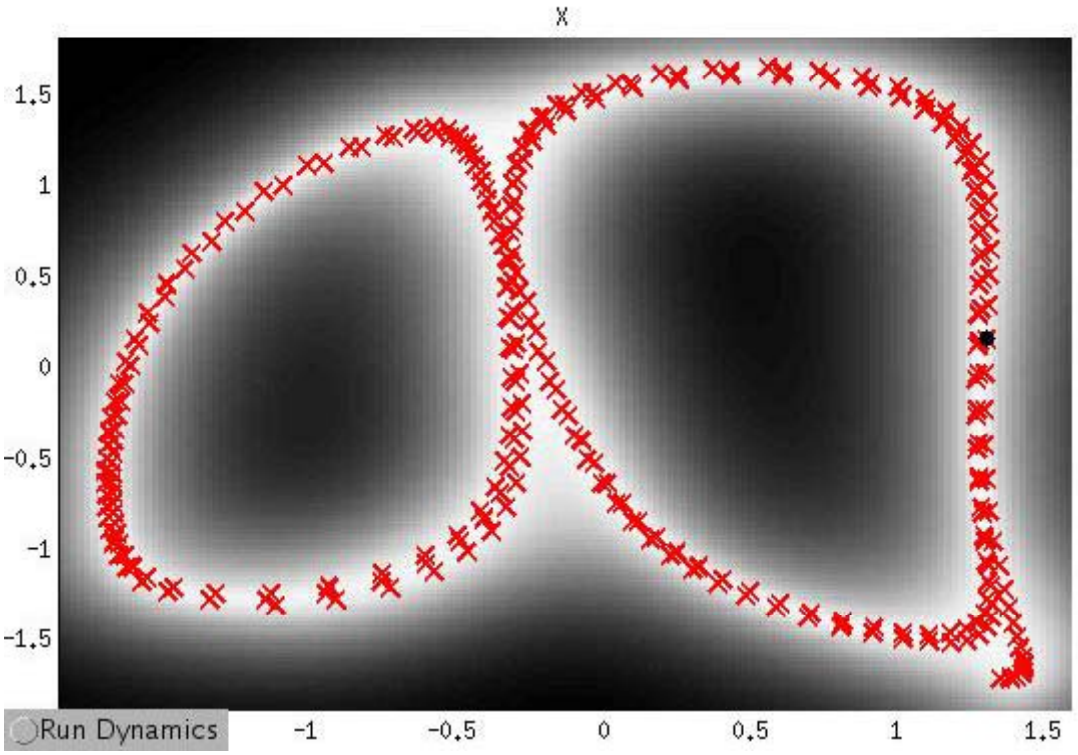
- Option for speedy optimisation:
  - Assume $y_t = \mu_y(x_t)$ -> optimise terms 1,3,4

Lifting the 3D Pose from a static image

- Example

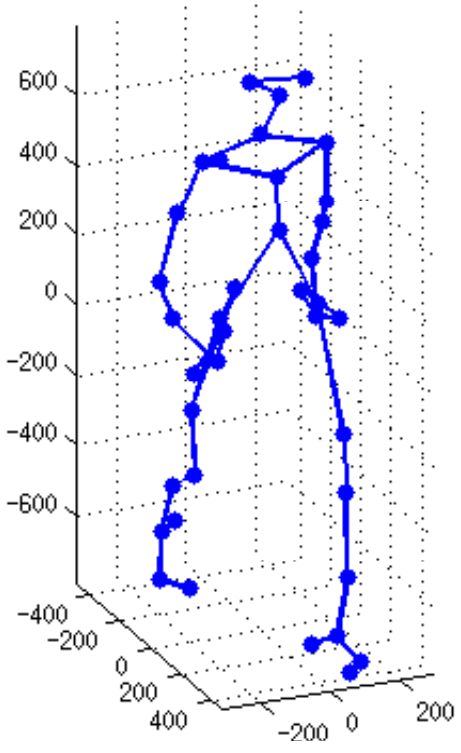
Lifting the 3D Pose from a static image

- Example – Latent Space (reconstructed latent variable)



Lifting the 3D Pose from a static image

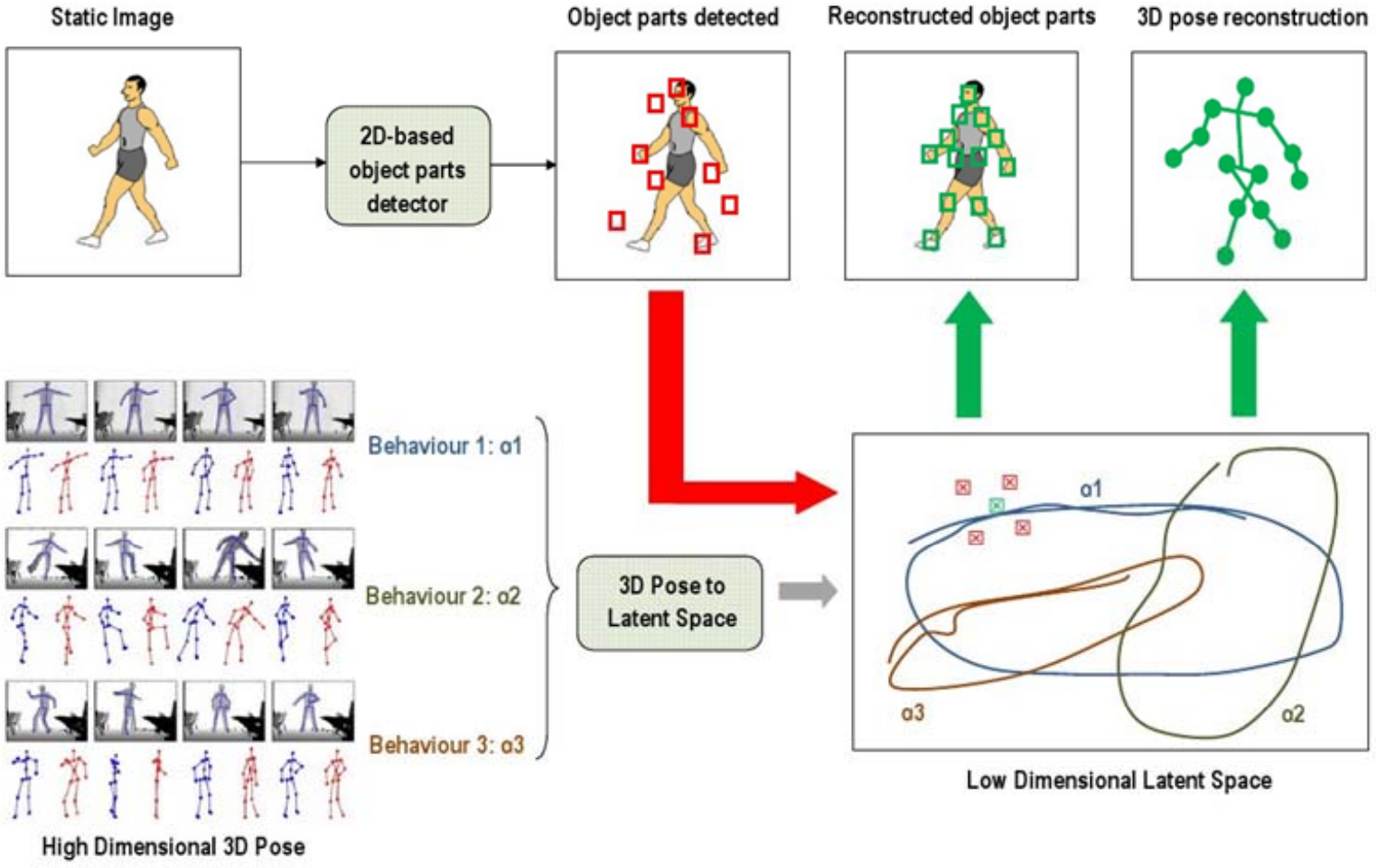
- Example – Reconstructed 3D pose



Steps

1. Dataset design and creation (3D & 2D).
2. Probabilistic learning of 3D pose.
3. Lifting the 3D pose from a static image.
4. **Linkage between 2D-based object recognition and 3D-based pose learning.**

Bridging the gap between 2D & 3D



Final Remarks

- Conclusions
  - Recovering 3D pose from a static image is feasible in a probabilistic framework using low dimensional latent spaces.
  - Experience based on massive learning of models, behaviours and subclasses is needed.
  - Real-time is not feasible at present.
  - A linkage can be established to bridge the gap between 2D-based object recognition techniques and 3D-based learning.
- Future work
  - General: everything is ahead (10% of expected progress at present).
  - Create massive dataset.
  - Full implementation of the linkage between 2D recognition and 3D learning based on latent spaces.

Muito obrigado!

For further information please contact:

Miguel Ángel Sotelo
miguel.sotelo@uah.es
www.robesafe.es/personal/sotelo