Object Pose Estimation summary

C. Papaioannidis, Prof. Ioannis Pitas
Aristotle University of Thessaloniki
pitas@csd.auth.gr
www.aiia.csd.auth.gr
Version 3.1.1

Artificial Intelligence & Information Analysis Lab

Object Pose Estimation

- **Introduction**
- 6D object pose estimation through object detection
- 3D object pose regression.
- 3D object pose classification.
- 3D object pose retrieval.

Applications and challenges

- Object pose estimation is a very challenging computer vision task.
- Heavily researched topic due to its importance in:
  - Robotics.
  - Augmented Reality.
- Challenges:
  - Occlusion.
  - Background clutter.
  - Scale and illumination variations.

Articulated object pose estimation

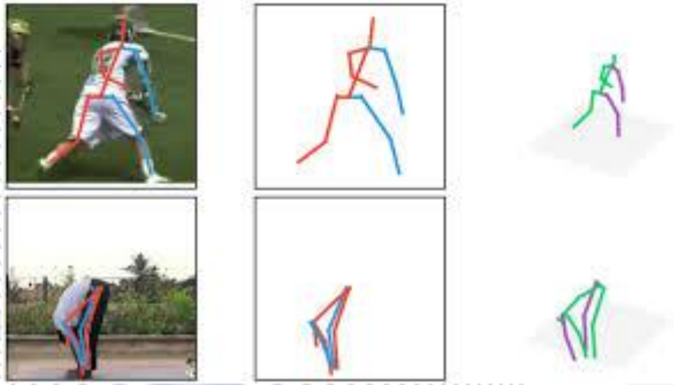
- There is a confusion on the use of terms pose and posture.
- ***Pose*** refers to the geometrical relation between an object and a camera.
- ***Posture*** refers to the spatial configuration of an articulated object.
- Human body posture estimation methods aim to estimate joint 2D or 3D coordinates (or the related angles).
  - Popular human pose/posture estimation methods:
    - OpenPose, DensePose.

Articulated object pose estimation

- It is a different problem than object pose estimation.
  - Joint angle estimation for the various joints.
  - Human Pose Estimation.

OpenPose.

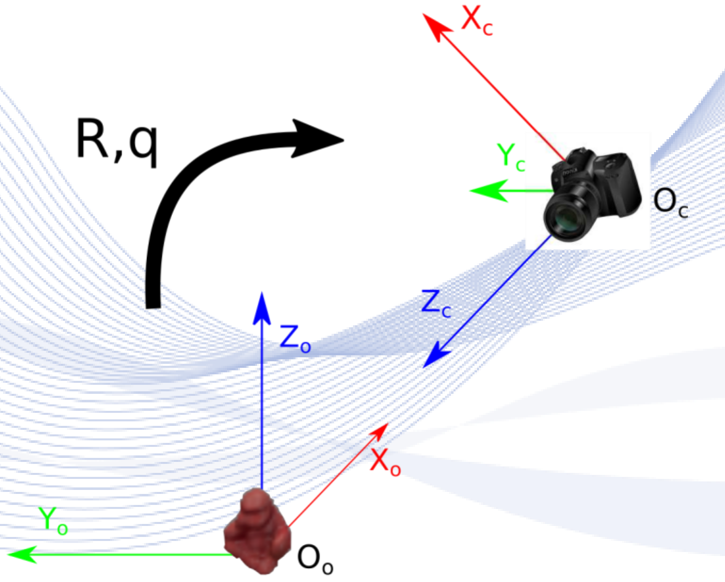
DensePose.



3D human pose.

Object pose estimation

- 3D Object Pose Estimation.
  - 3D Rotation matrix estimation.
- Object orientation in a camera coordinate system.
- Challenging computer vision task.
- Sub-case of 6D Object Pose Estimation.



Facial Pose Estimation

- Special case of object pose estimation.
  - Important in human-centered computing.
  - Facial Pose Estimation (regression)
  - Facial Pose Classification (e.g., frontal, side pose).

Facial Pose Datasets

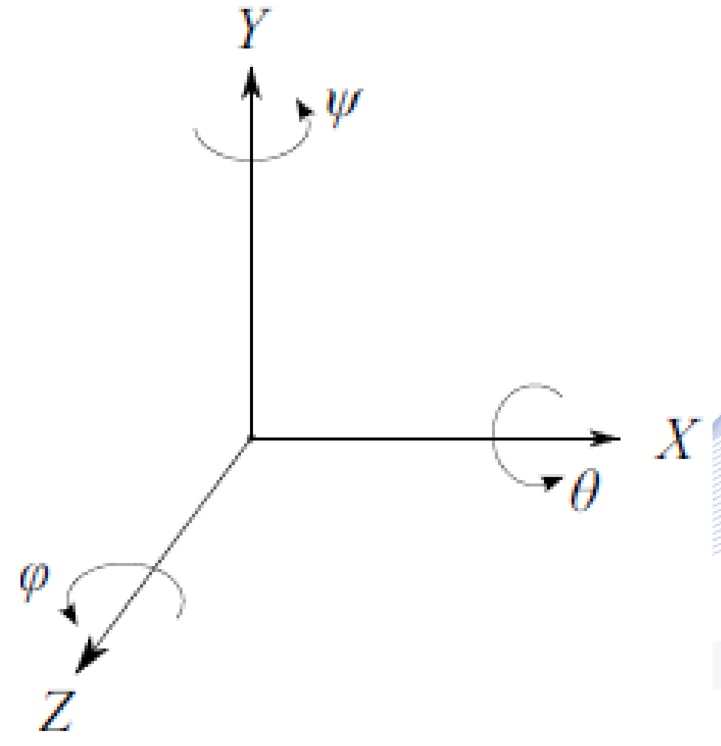
- Two datasets used for evaluating the proposed method:
- Annotated Facial Landmarks in the Wild dataset (AFLW):
  - Continuous horizontal pose annotations.

Facial Pose Datasets

- Head Pose Image Dataset (HPID)
  - 13 discrete horizontal pose annotations

3D rotation representations

- An arbitrary rotation in the 3D space can be represented by the Euler rotation angles θ, ψ, ϕ about the X, Y, Z axes.



3D rotation representations

- Matrix representation of clockwise rotation about each X, Y, Z axis:

$$\mathbf{R} = \mathbf{R}_z\mathbf{R}_y\mathbf{R}_x = \begin{bmatrix} \cos\phi\cos\psi & \cos\phi\sin\psi\sin\theta - \sin\phi\cos\theta & \cos\phi\sin\psi\cos\theta + \sin\phi\sin\theta \\ \sin\phi\cos\psi & \sin\phi\sin\psi\sin\theta + \cos\phi\cos\theta & \sin\phi\sin\psi\cos\theta - \cos\phi\sin\theta \\ -\sin\psi & \cos\psi\sin\theta & \cos\psi\cos\theta \end{bmatrix}$$

$$\mathbf{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \qquad \mathbf{R}_y = \begin{bmatrix} \cos\psi & 0 & \sin\psi \\ 0 & 1 & 0 \\ -\sin\psi & 0 & \cos\psi \end{bmatrix} \qquad \mathbf{R}_z = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- The order of matrices in this equation does matter.

3D rotation representations

- 3D rotation can also be represented by *quaternions* that are extensions of complex numbers:

$$\mathbf{q} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$$

q_0, q_1, q_2, q_3 are real numbers and:

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$$

- Unit quaternion $\mathbf{q}_R = [q_0\ q_1\ q_2\ q_3]^T$. It satisfies:

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1.$$

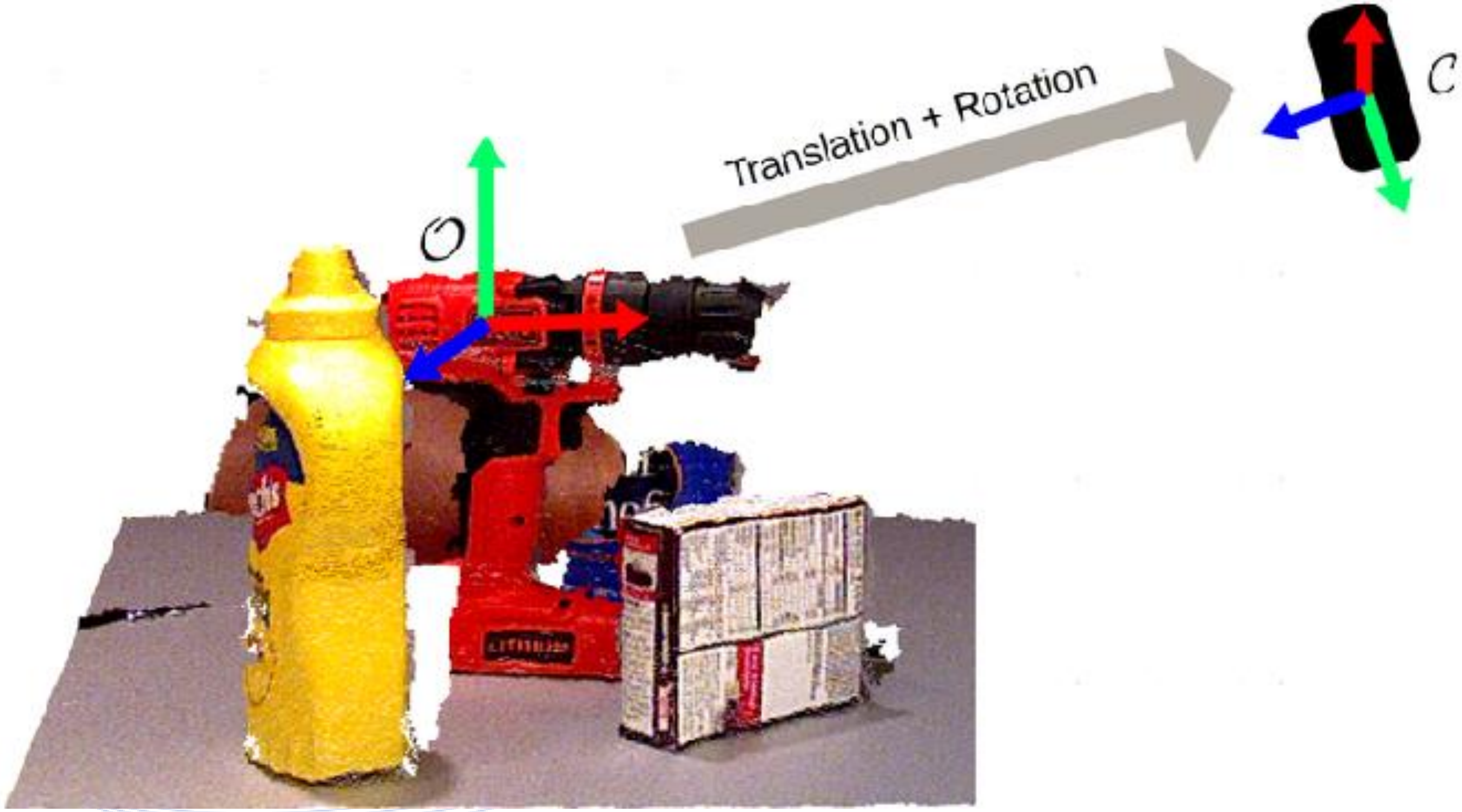
6D object pose estimation

- Estimate object coordinate system orientation and translation relative to the camera coordinate system.
- Object orientation is usually represented by a rotation matrix $\mathbf{R} \in \mathbb{R}^{3\times3}$, where:

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}.$$

- Object translation is estimated in the form of a translation vector $\mathbf{T} \in \mathbb{R}^3$, $\quad \mathbf{T} = [T_X, T_Y, T_Z]^T$.

6D object pose estimation



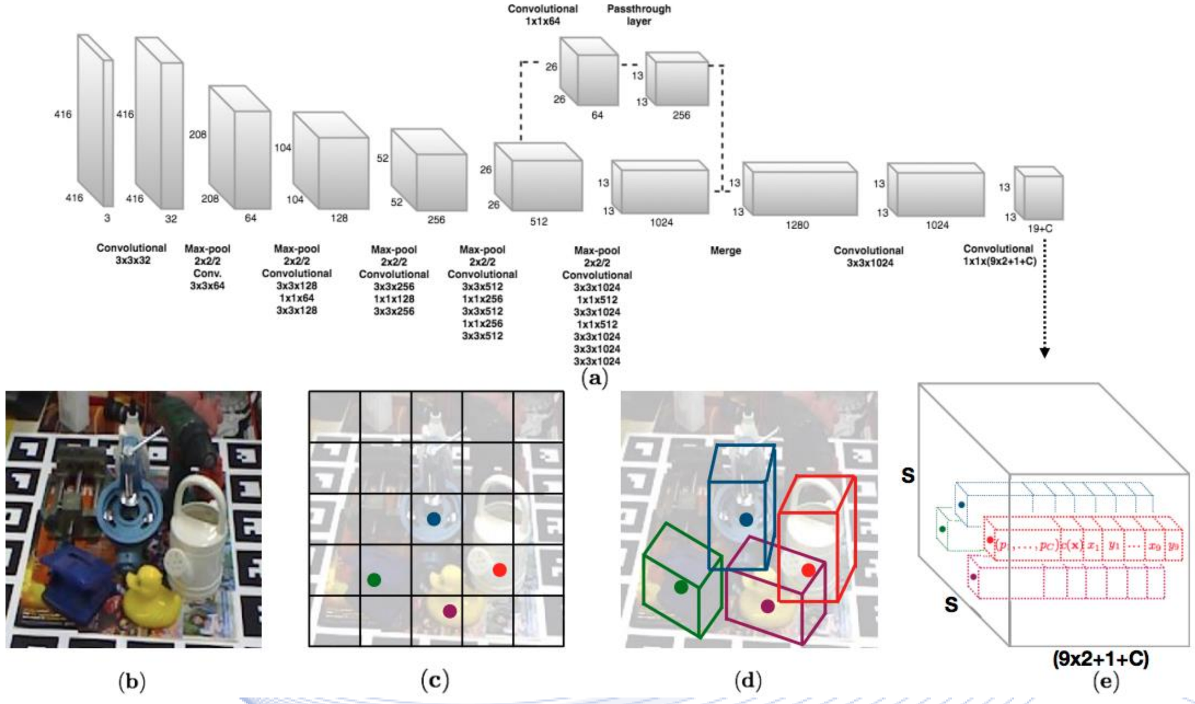
Object Pose Estimation

- Introduction
- **6D object pose estimation through object detection**
- 3D object pose regression.
- 3D object pose classification.
- 3D object pose retrieval.

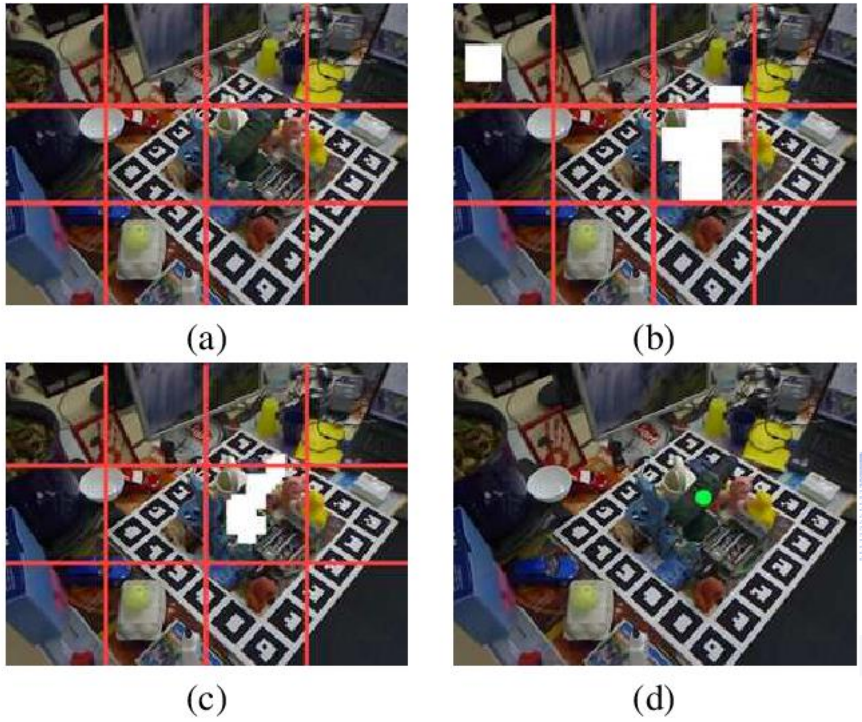
DNN Regression for Object Pose Estimation

- Machine Learning Approach
  - A neural network receives the object image and directly regresses its pose.
  - Only a set of pose-annotated object pictures are needed:
    - There is no need to manually develop 3-D models.
    - The models are more robust to variations of the object for which we want to estimate its pose.
    - The pose estimation can run entirely on GPU and (possibly) incorporated into a unified detection+pose estimation neural network.
  - Very few pre-trained models are available:
    - Models must be trained for the objects of interest (faces, bicycles, boats, etc.).

6D object pose estimation using deep object detection



Object detection.

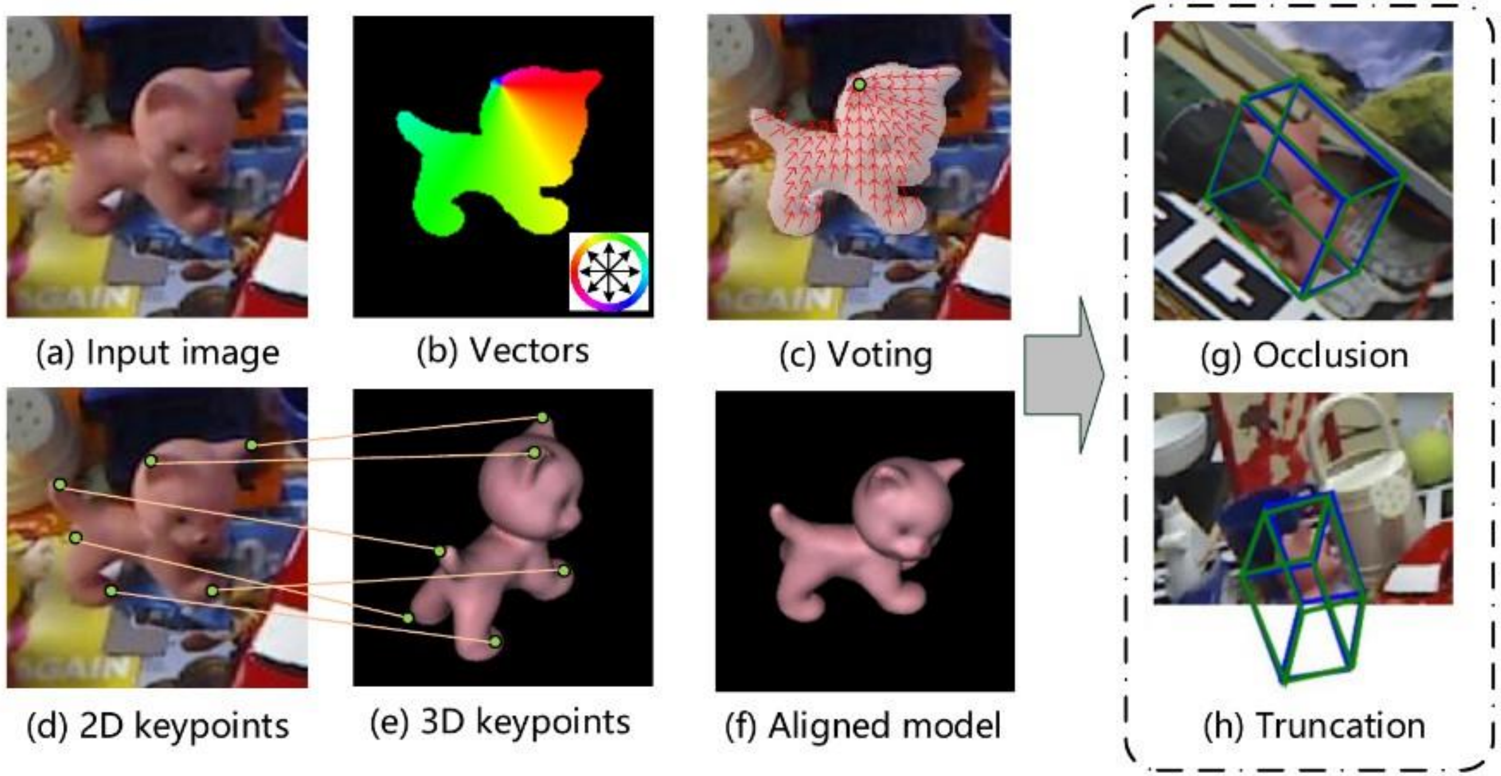


Segmentation.

6D object pose estimation using deep object detection

- The 2D object detections are given as inputs to the pose estimation step.
- Given:
  - the camera intrinsic parameters,
  - the 3D coordinates of the object predefined keypoints or bounding box corners in the object coordinate system,
- the 6D object pose is calculated from the correspondences between the 2D and 3D points using a ***Perspective-n-Point*** (***PnP***) algorithm.

6D object pose estimation using deep object detection



6D object pose estimation with 2D keypoint detection.

Object Pose Estimation

- Introduction
- 6D object pose estimation through object detection
- 3D object pose regression.
- 3D object pose classification.
- 3D object pose retrieval.

3D object pose estimation

- Many CNN-based methods have been proposed for the 3D object pose estimation step.
- Main method categories:
  - **3D object pose regression.**
  - **3D object pose classification.**
  - 3D object pose retrieval.

3D object pose regression

- Given an image $\mathbf{x}$ depicting an object at a specific 3D pose, 3D object pose regression methods aim to directly regress its 3D pose $\mathbf{p}$ through a simple CNN forward pass:

$$\hat{\mathbf{p}} = \boldsymbol{f}(\mathbf{x}; \boldsymbol{\theta}),$$

- $\boldsymbol{f}(\mathbf{x}; \boldsymbol{\theta})$: CNN having parameter vector $\boldsymbol{\theta}$.
- Pre-trained CNNs or a separate CNN for each object of interest are required.
- 3D object pose predictions lack increased accuracy.

3D object pose classification

- The continuous 3D pose space is quantized to a predefined number of orientation classes $\mathbf{p}_i$.
- Similar to 3D object pose regression, ***3D object pose classification*** methods aim to classify an object image $\mathbf{x}$ to its orientation class $\mathbf{p}_i$ through a simple network pass:

$$\hat{\mathbf{p}}_i = \boldsymbol{f}(\mathbf{x}; \boldsymbol{\theta}).$$

- Pre-trained CNNs or a separate CNN for each object of interest are also required.
- Increased accuracy relative to regression methods.

3D object pose retrieval

- These methods aim to extract 3D pose-related image features using CNNs.
- A codebook is constructed, consisting of images depicting objects at a predefined number of different 3D poses that cover the 3D pose space.
- The 3D object pose is estimated by matching a test object image with the most similar codebook image and returning its corresponding ground truth 3D pose.

Object Pose Estimation

- Introduction
- 6D object pose estimation through object detection
- 3D object pose regression.
- 3D object pose classification.
- **3D object pose retrieval.**

Artificial Intelligence & Information Analysis Lab

3D object pose retrieval

- A CNN is trained to extract 3D pose-related features.
- Using the trained CNN, ***codebook features*** $\mathbf{f}_{c_i}, i = 1, \dots, m$ are first calculated offline and stored:

$$\mathbf{f}_{c_i} = \boldsymbol{f}\left(\mathbf{x}_{c_i}; \boldsymbol{\theta}\right).$$

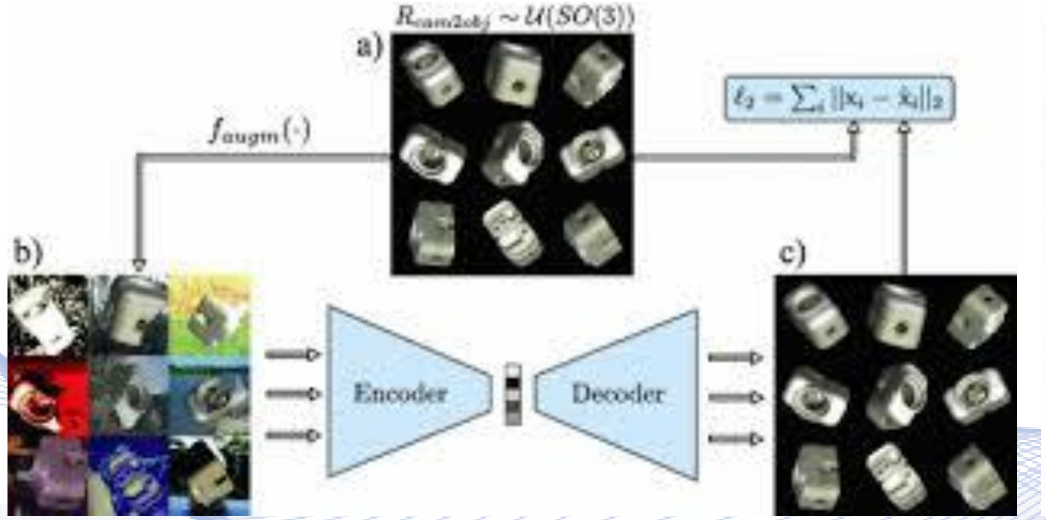
- Given a test object image $\mathbf{x}$, the corresponding feature vector is extracted using the same trained CNN:

$$\mathbf{f} = \boldsymbol{f}(\mathbf{x}; \boldsymbol{\theta}).$$

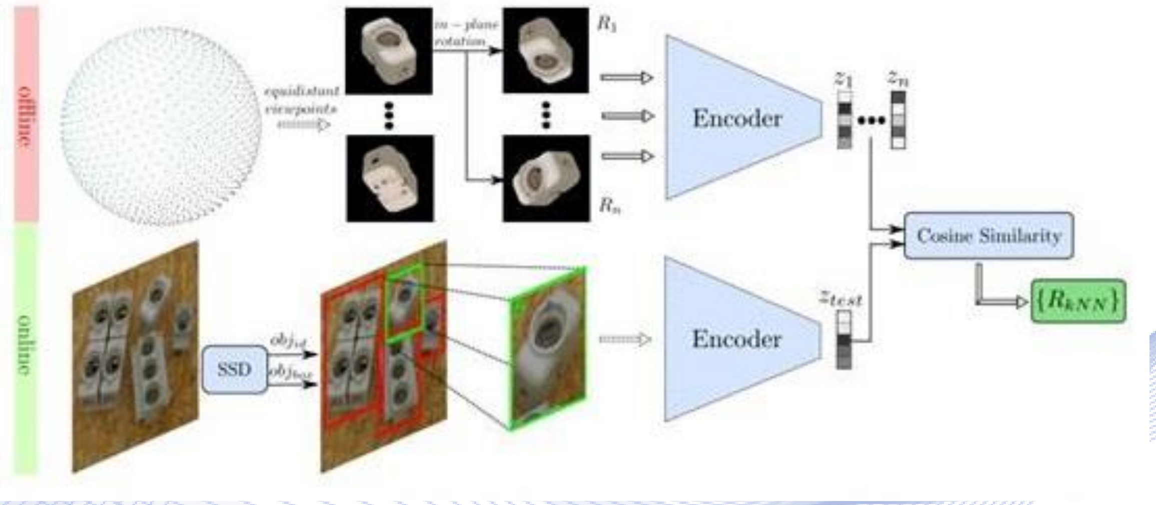
- The extracted test image feature vector $\mathbf{f}$ is matched to the most similar $\mathbf{f}_{c_i}, i = 1, \dots, m$ using a matching algorithm (Nearest Neighbor) and the corresponding ground truth 3D pose is returned as the 3D pose estimate.

Artificial Intelligence & Information Analysis Lab

Learning 3D pose features using autoencoders



Training.



Testing.

Quaternion learning for 3D object pose estimation

- Distance metric between two rotation matrices $\mathbf{R}_i, \mathbf{R}_j$:

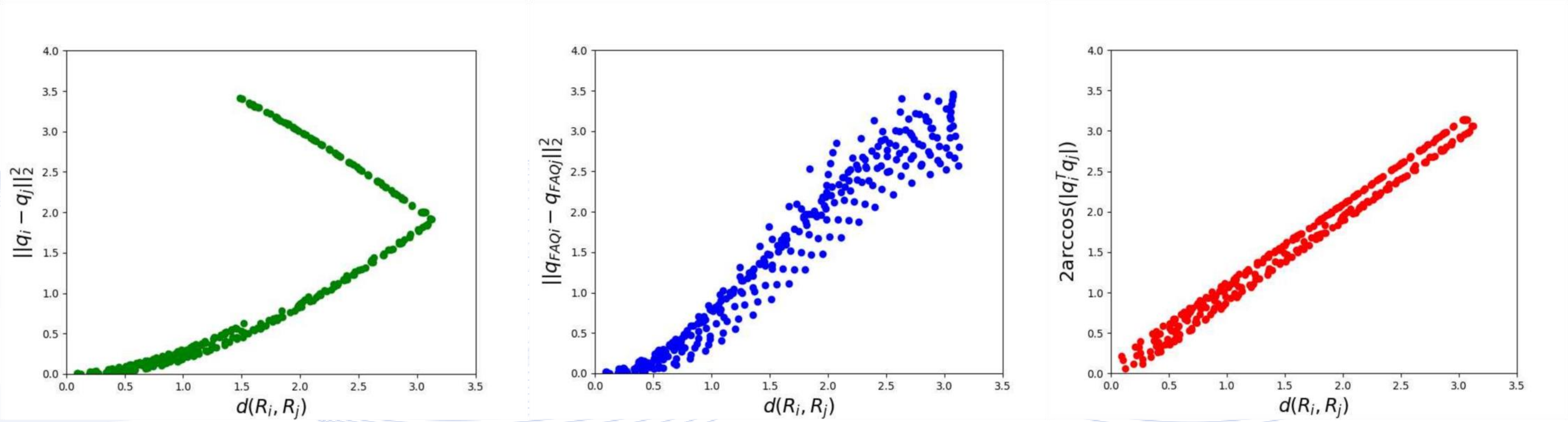
$$d(\mathbf{R}_i, \mathbf{R}_j) = \left\|\log(\mathbf{R}_i{}^T \mathbf{R}_j)\right\|_2.$$

Different quaternion distance metrics were investigated to find the one the best resembles $d(\mathbf{R}_i, \mathbf{R}_j)$.

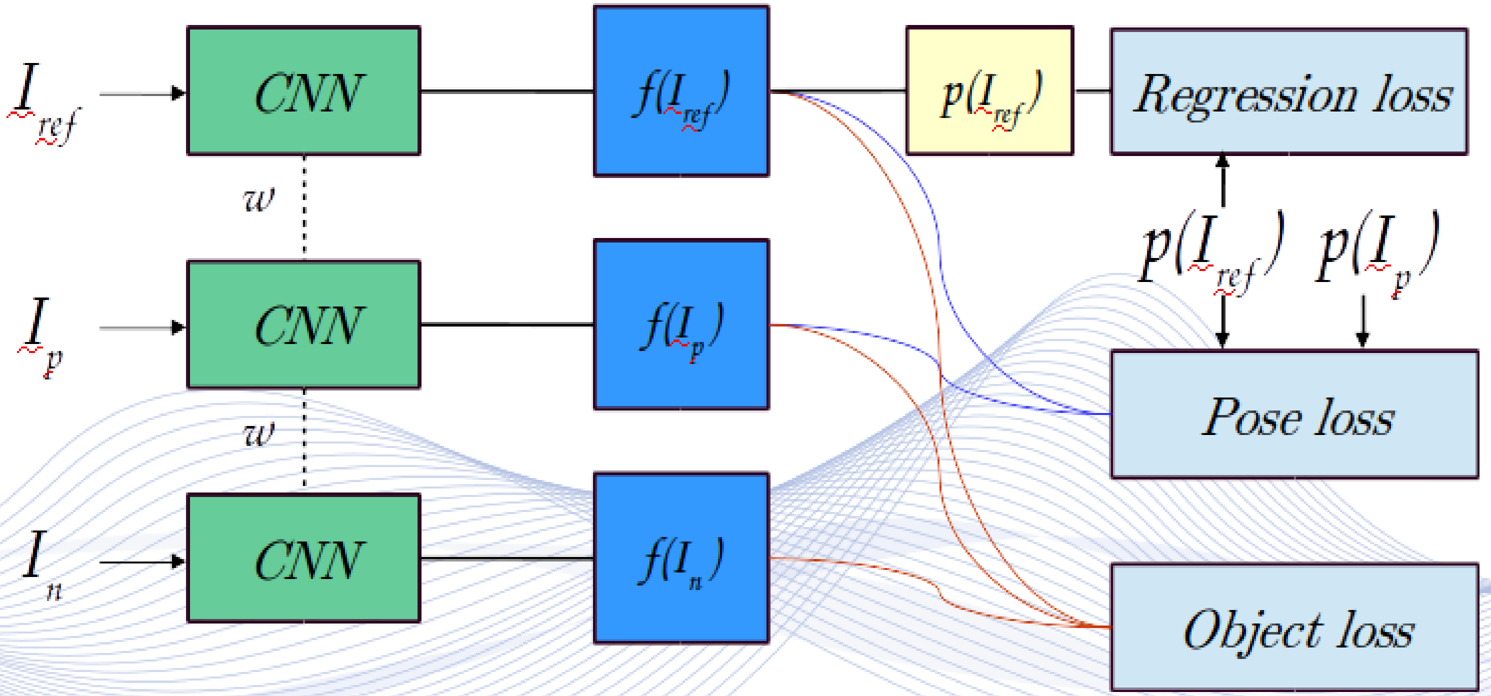
- Squared Euclidean distance: $d_E = \left\|\mathbf{q}_i - \mathbf{q}_j\right\|_2^2$.
- Full angle Quaternion distance: $d_C = \left\|\mathbf{q}_{faq_i} - \mathbf{q}_{faq_j}\right\|_2^2$.
- ***Inverse cosine distance***: $d_{IC} = 2\arccos(|\mathbf{q}_i^T \mathbf{q}_j|)$.

Quaternion learning for 3D object pose estimation

- The inverse cosine distance was selected as it has a linear relationship with $d(\mathbf{R}_i, \mathbf{R}_j)$.



Quaternion learning for 3D object pose estimation



Quaternion learning for 3D object pose estimation

- Objective loss function:

$$J = J_{desc} + J_{qreg} + \lambda\|w\|_2^2.$$

- Error J_{desc} aims to learn pose features from object images:

$$J_{desc} = J_p + J_o.$$

- Pairwise loss between images of the same object:

$$J_p = \sum_{s_i, s_j} \{\|\mathbf{f}_i - \mathbf{f}_j\|_2^2 - 2\arccos(|\mathbf{q}_i^T \mathbf{q}_j|)\}^2.$$

Artificial Intelligence & Information Analysis Lab

Quaternion learning for 3D object pose estimation

- Triplet loss between images of the same object and an image of a different object:

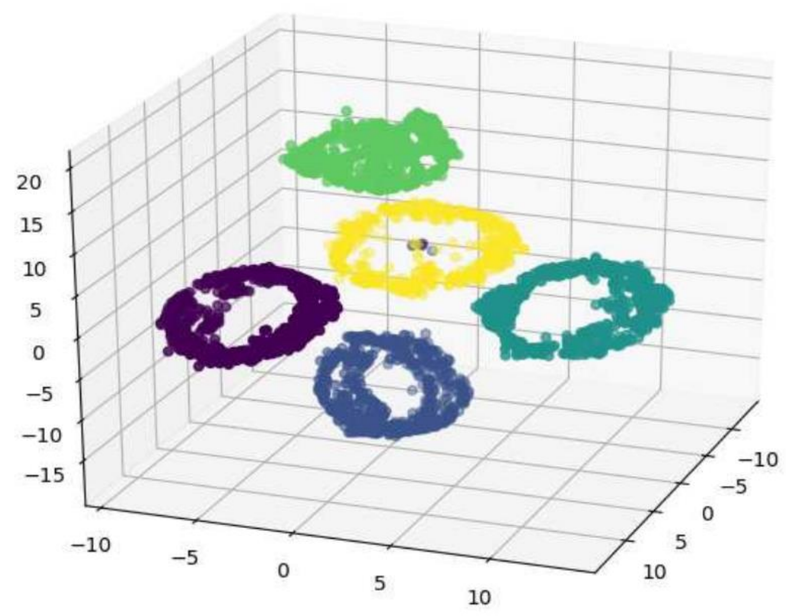
$$J_o = \sum_{s_i, s_j, s_k} \frac{\left\|\mathbf{f}_i - \mathbf{f}_j\right\|_2}{\|\mathbf{f}_i - \mathbf{f}_k\|_2 + \varepsilon}.$$

- Quaternion regression loss:

$$J_{qreg} = \|\mathbf{q} - \hat{\mathbf{q}}\|_2^2.$$

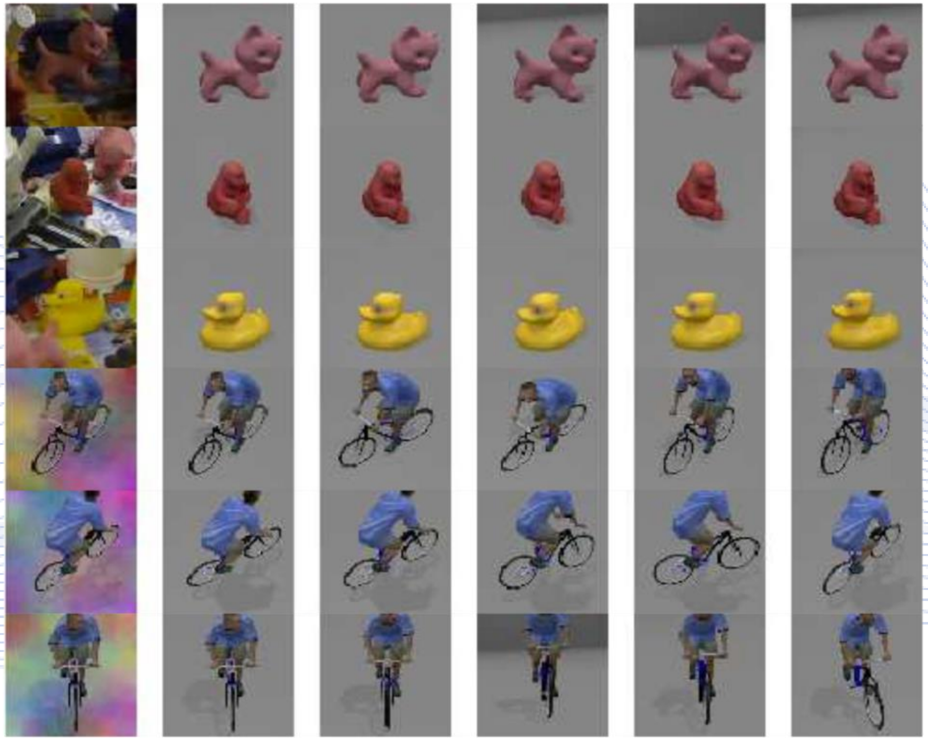
Quaternion learning for 3D object pose estimation

- Visualization of the learned features for 5 random objects from the LineMod dataset.



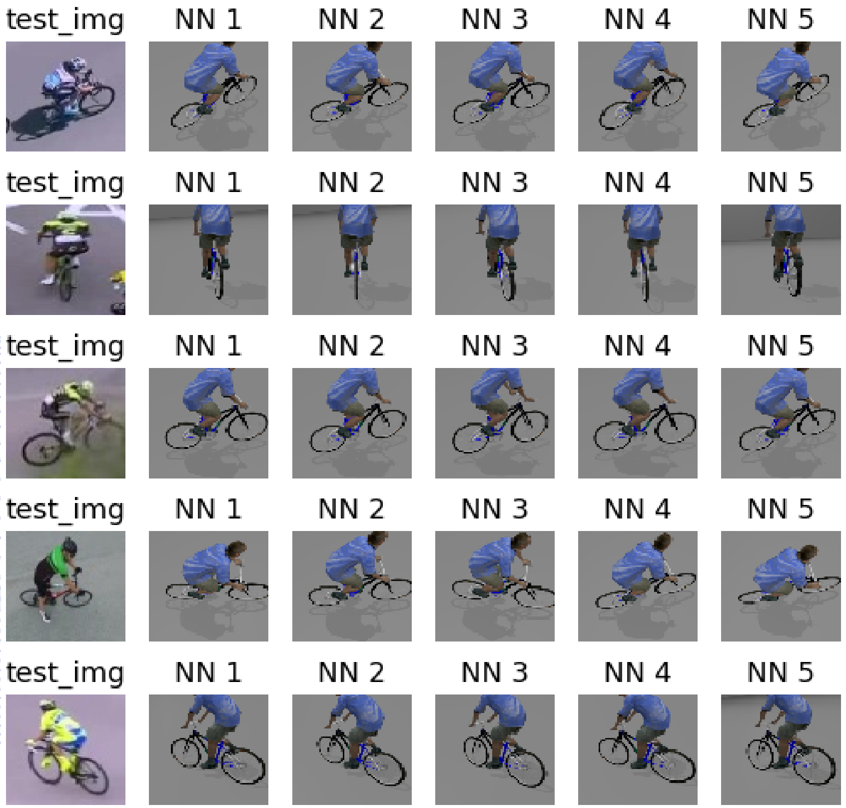
Quaternion learning for 3D object pose estimation

- 5 retrieved templates that are closest to the test image in the feature space.



Quaternion learning for 3D object pose estimation

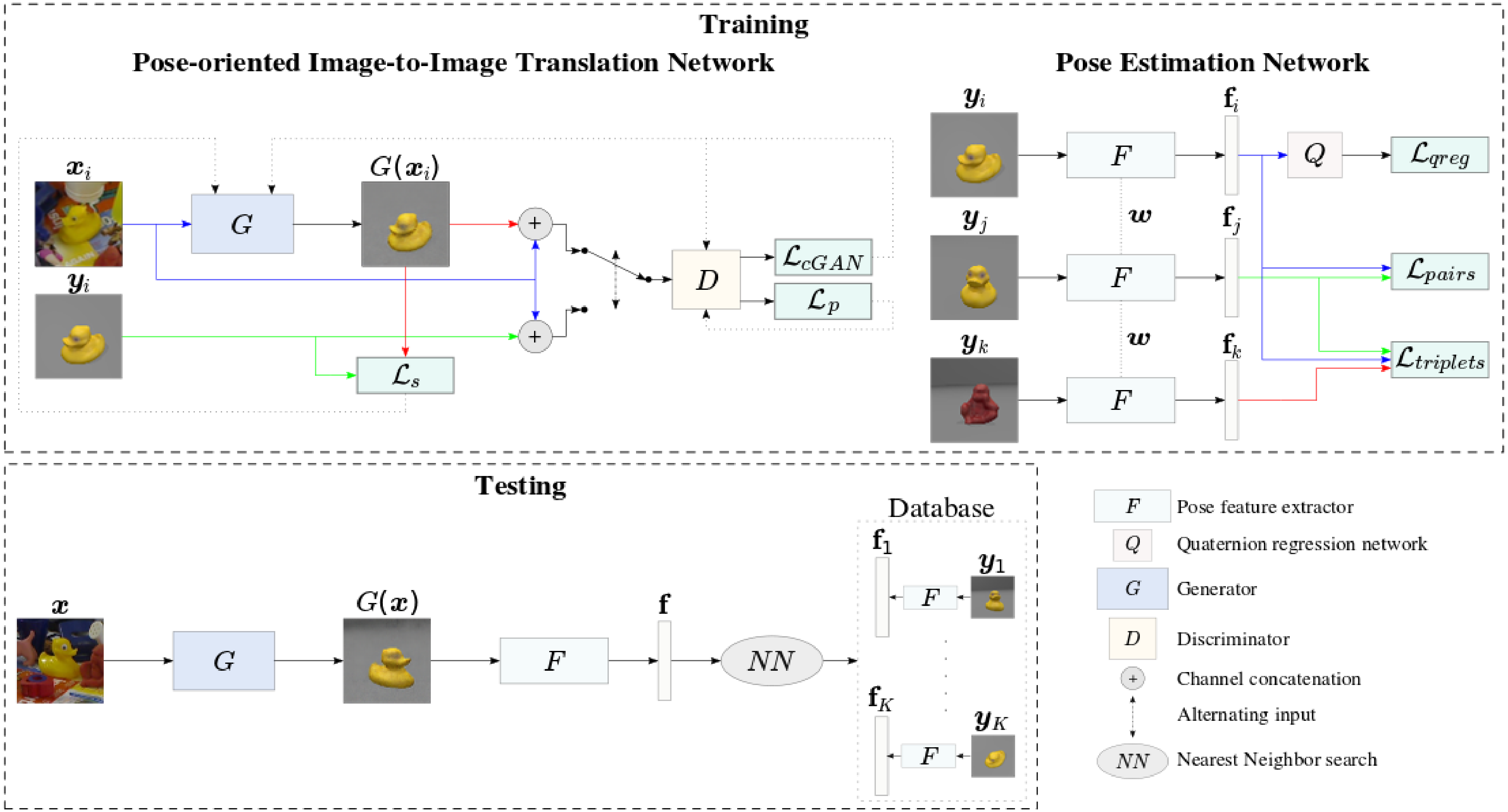
- 5 retrieved templates that are closest to real cyclist test images.



Domain-translated object pose estimation

- ***Image-to-image translation*** step to translate real images to synthetic ones.
- Pose estimation step applied on the translated synthetic images.
- Translated synthetic images are clear from redundant noise.
- 3D poses can be more accurately predicted from the noise-free translated images.

Domain-translated object pose estimation



Domain-translated object pose estimation

- 5 retrieved templates that are closest to real cyclist test images.

References

1. C. Papaioannidis and I. Pitas, “3D Object Pose Estimation using Multi-Objective Quaternion Learning”, 2019.
2. C. Papaioannidis, V. Mygdalis and I. Pitas, “Domain-Translated 3D Object Pose Estimation”, 2019.
3. Z. Cao, et al., “OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields”, 2018.
4. R. A. Guler, N. Neverona and I. Kokkinos, “DensePose: Dense Human Pose Estimation In The Wild”, 2018.
5. J. Martinez, R. Hossain, J. Romero and J. J. Little, “A simple yet effective baseline for 3d human pose estimation”, 2017.
6. G. Gao, M. Lauri, J. Zhang and S. Frintrop, “Occlusion Resistant Object Rotation Regression from Point Cloud Segments”, 2018.
7. W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab, “SSD-6D: Making RGB-Based 3D Detection and 6D Pose Estimation Great Again”, 2017.
8. B. Tekin, S. N. Sinha and P. Fua, “Real-Time Seamless Single Shot 6D Object Pose Estimation”, 2018.
9. M. Rad and V. Lepetit, “BB8: A Scalable, Accurate, Robust to Partial Occlusion Method for Predicting the 3D Poses of Challenging Object without Using Depth”, 2017.
10. M. Sundermeyer, et al., “Implicit 3D Orientation Learning for 6D Object Detection from RGB Images”, 2018.

Q & A

Thank you very much for your attention!

More material in
http://icarus.csd.auth.gr/cvml-web-lecture-series/

Contact: Prof. I. Pitas
pitas@csd.auth.gr