Deep Learning algorithms for intelligent support of workers

Dr. Christos Papaioannids, Prof. Ioannis Pitas
Aristotle University of Thessaloniki
pitas@csd.auth.gr
www.aiia.csd.auth.gr
Version 1.0

Artificial Intelligence &
Information Analysis Lab

Contents

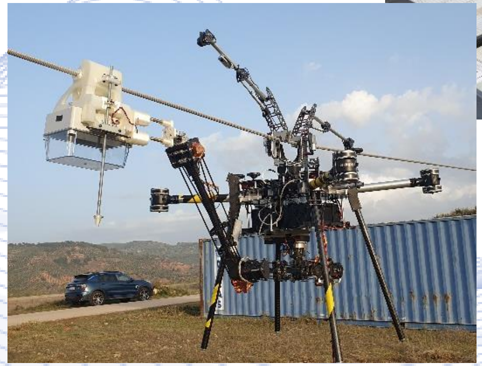
- Introduction
- Deep Neural Networks (DNNs)
- 2D object detection and tracking
- Semantic image segmentation
- Visual anomaly detection
- Human pose estimation
- Action/gesture recognition
- Applications

Contents

- Introduction
- Deep Neural Networks (DNNs)
- 2D object detection and tracking
- Semantic image segmentation
- Visual anomaly detection
- Human pose estimation
- Action/gesture recognition
- Applications

Introduction

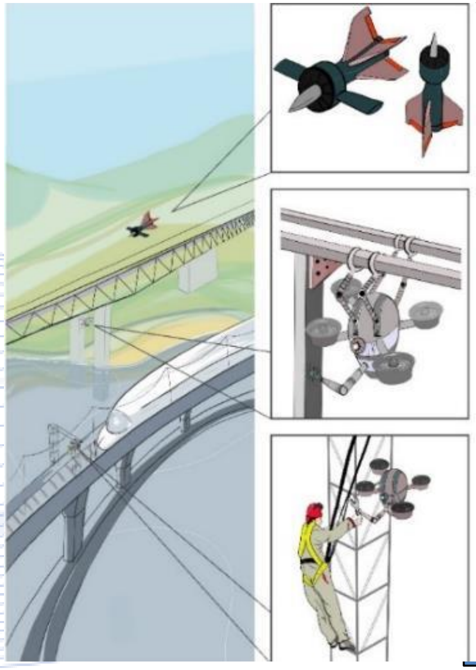
- Deep learning-based algorithms allow the development of advanced autonomous systems that can:
  - understand their surrounding environment,
  - make decisions,
  - perform simple and complex tasks.



- Benefits for human workers:
  - increased safety,
  - increased efficiency,
  - reduced workload and stress.
- Examples: industrial robots, autonomous UAVs (drones), etc.

Introduction

- Application example: inspection and maintenance of large infrastructures via an aerial cognitive robotic system.



Long range and local very accurate inspection of the infrastructure

Maintenance activities based on aerial manipulation

Safe Aerial co-working

Introduction

- Powerline infrastructure inspection and maintenance.
  - EU: About 5 million km.
  - Inspections performed by crewed helicopters → risk of workers.
  - Cost: ~150€/km.
- Benefits:
  - Safety of workers.
  - Reduced cost and sustainability.

Introduction

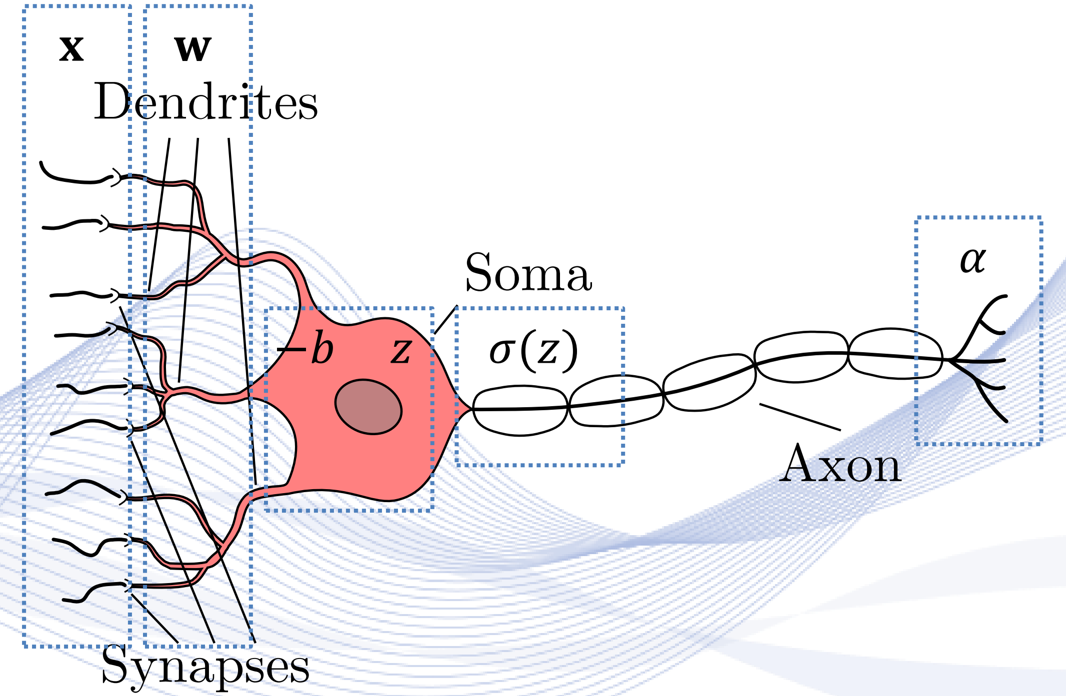
- Pipeline infrastructure inspection.
  - Oil & Gas facilities.
  - Degradation of materials due to environmental exposure and mechanical demand.
- Benefits:
  - Safety of workers.
  - Reduced workload and stress.

Contents

- Introduction
- Deep Neural Networks (DNNs)
- 2D object detection and tracking
- Semantic image segmentation
- Visual anomaly detection
- Human pose estimation
- Action/gesture recognition
- Applications

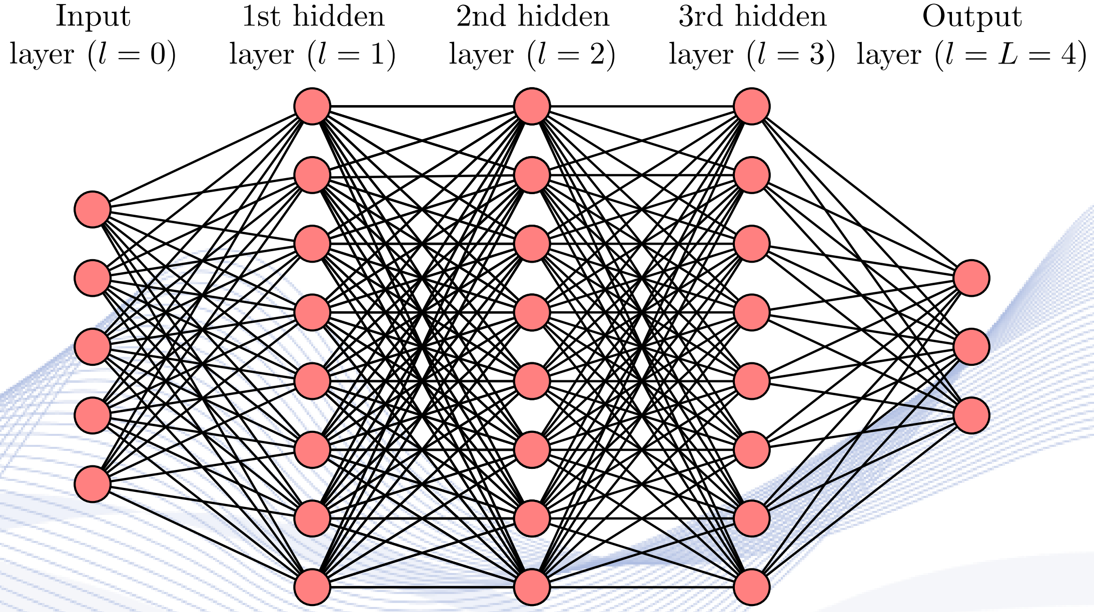
Multi-Layer Perceptron

- Perceptron:
  - Simplest mathematical model of a biological neuron.
  - Real inputs $\mathbf{x}, x_i \in [0,1]$.
  - Activation $\alpha \in \{0,1\}$.
  - Activation function $\sigma(\cdot)$.
  - Firing threshold: $\mathbf{w}^T\mathbf{x} \geq -b$.
  - $\alpha = \sigma(z) = \sigma(\mathbf{w}^T\mathbf{x} + b)$.



Multi-Layer Perceptron

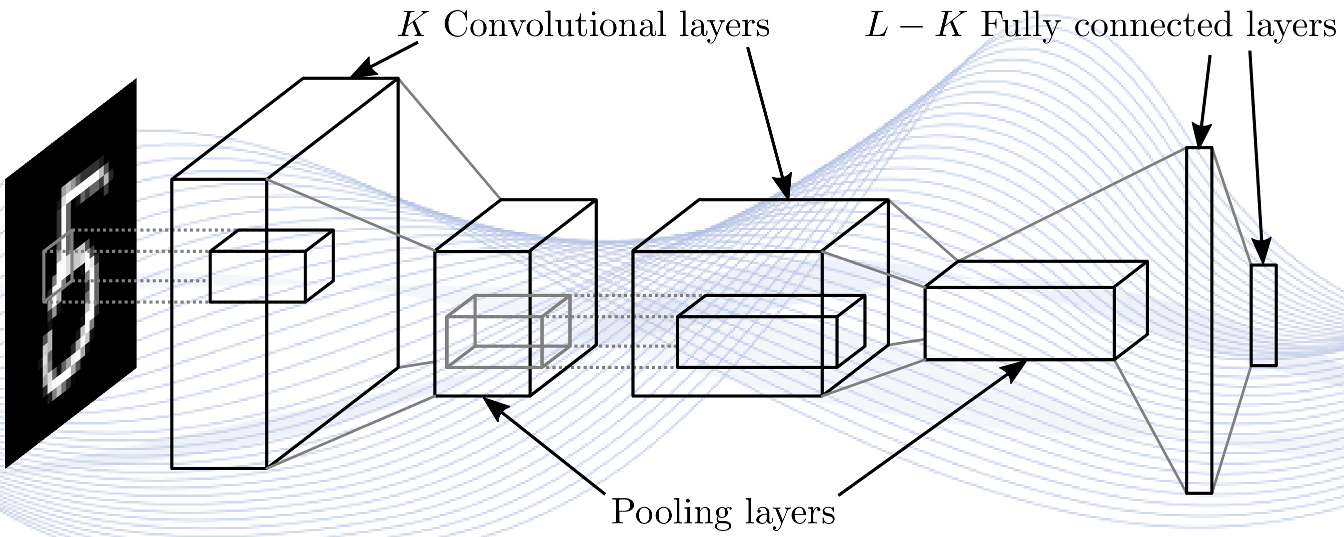
- Multi-Layer Perceptron (MLP):
  - Multiple layers L, with multiple neurons $n_l, l = 1, \dots, L$.
  - The input layer ($l = 0$) has k inputs. k: dimensionality of the input $\mathbf{x}$.
  - The $L-1$ hidden layers $l = 1, \dots, L-1$ may have any number of neurons.
  - The output layer $l = L = 4$ should match the dimensionality of the desired final output $\mathbf{y}$.



MLP with $L = 4$ layers.

Convolutional Neural Networks

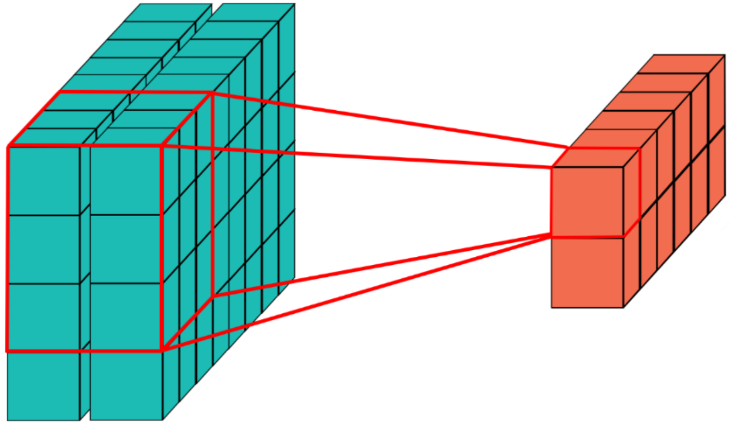
- RGB images cannot be processed by MLPs efficiently, due to the increased number of input features: $k = H \times W \times 3$.
- Convolutional Neural Networks (CNNs) → weight sharing.



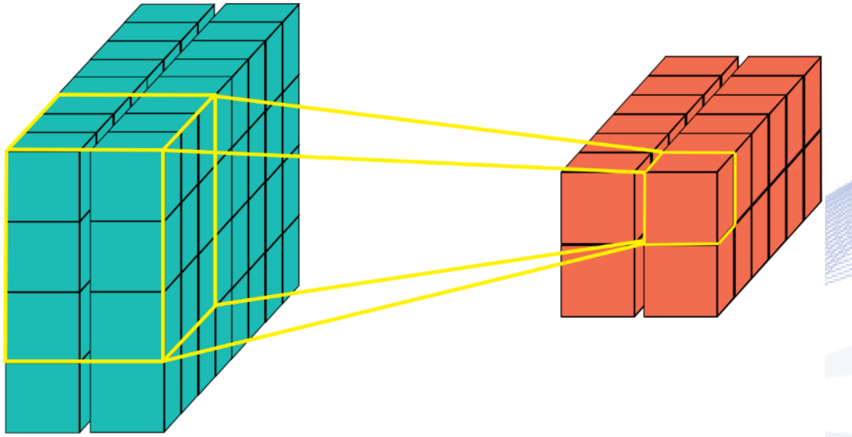
Simple CNN architecture.

Convolutional Neural Networks

- 2D convolutional layers:
  - Convolution operation.
  - 3D kernels/filters.



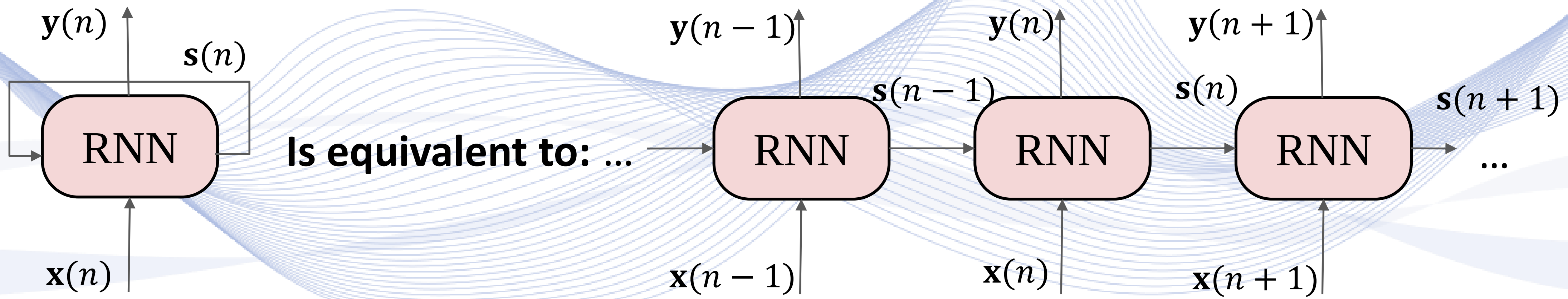
Convolution with a single 3 × 3 × 2 kernel/filter.



Convolution with two 3 × 3 × 2 kernels/filters.

Recurrent Neural Networks

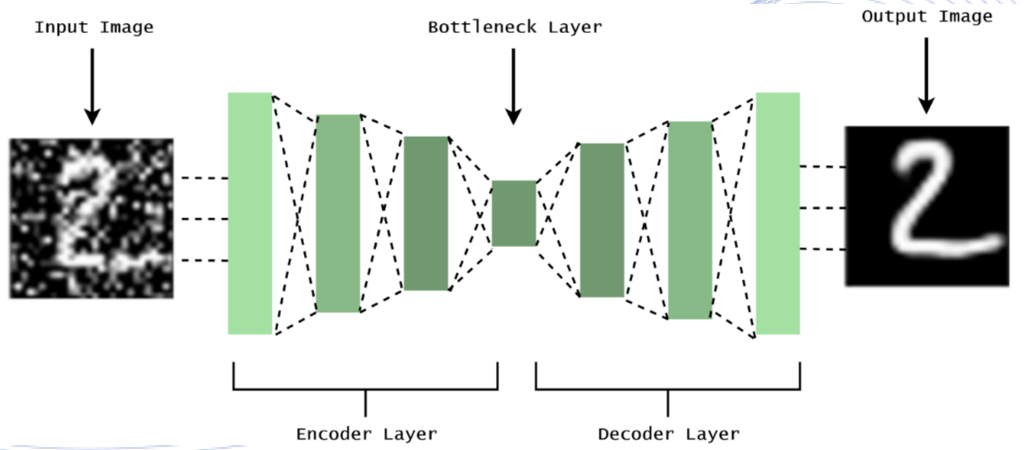
- Recurrent neural networks (RNNs):
  - Process sequential data (e.g., text, video).
  - Utilize information from previous time steps.
  - Advanced types of RNNs: Long Short-Term Memory networks (LSTMs), Gated Recurrent Units (GRUs), other.



Unfolding of an RNN with one recurrent layer through time.

Encoder-decoder networks

- Encoder-decoder networks consists of two networks: the encoder and the decoder.
  - Encoder and decoder: any DNN type (MLPs, CNNs, other).
  - Goal: extract rich input representations (code) or/and produce high-dimensional outputs.



Simple denoising autoencoder.

Encoder-decoder networks

- If output $\mathbf{y}$ is the same as the input $\mathbf{x}$: autoencoder.
- Encoder-decoder networks can also be used for data generation.

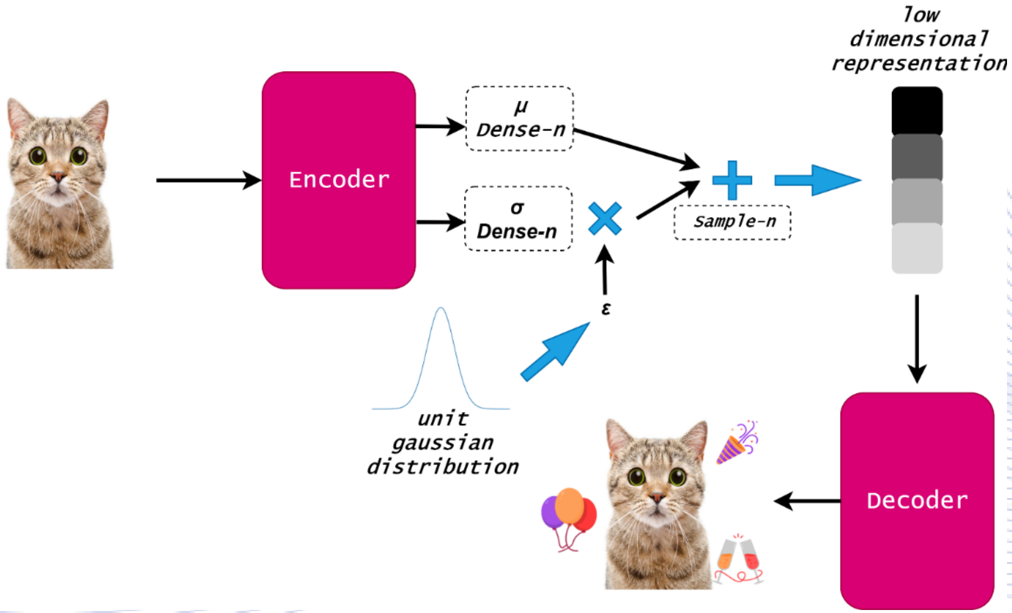
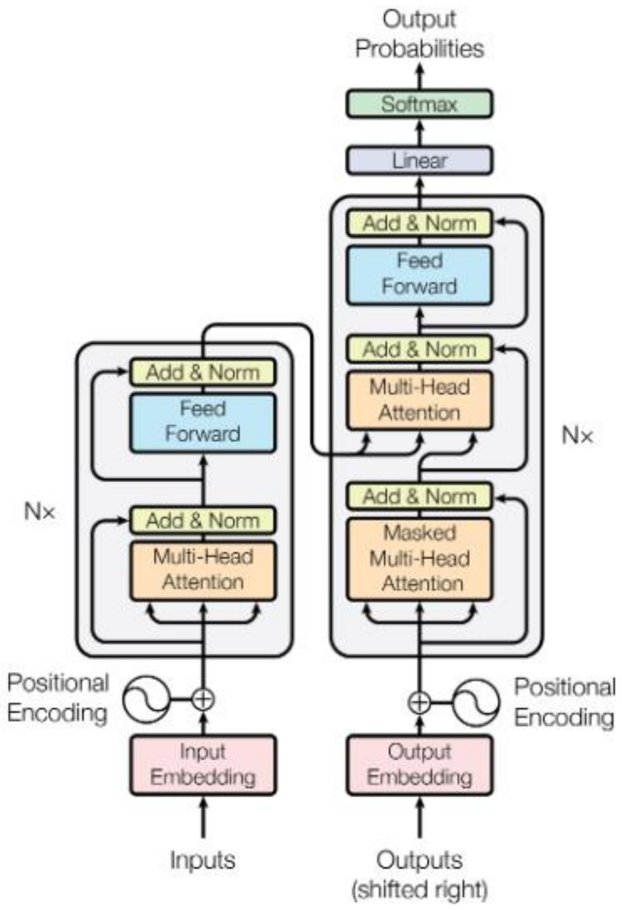


Image generation with an encoder-decoder.

Transformers

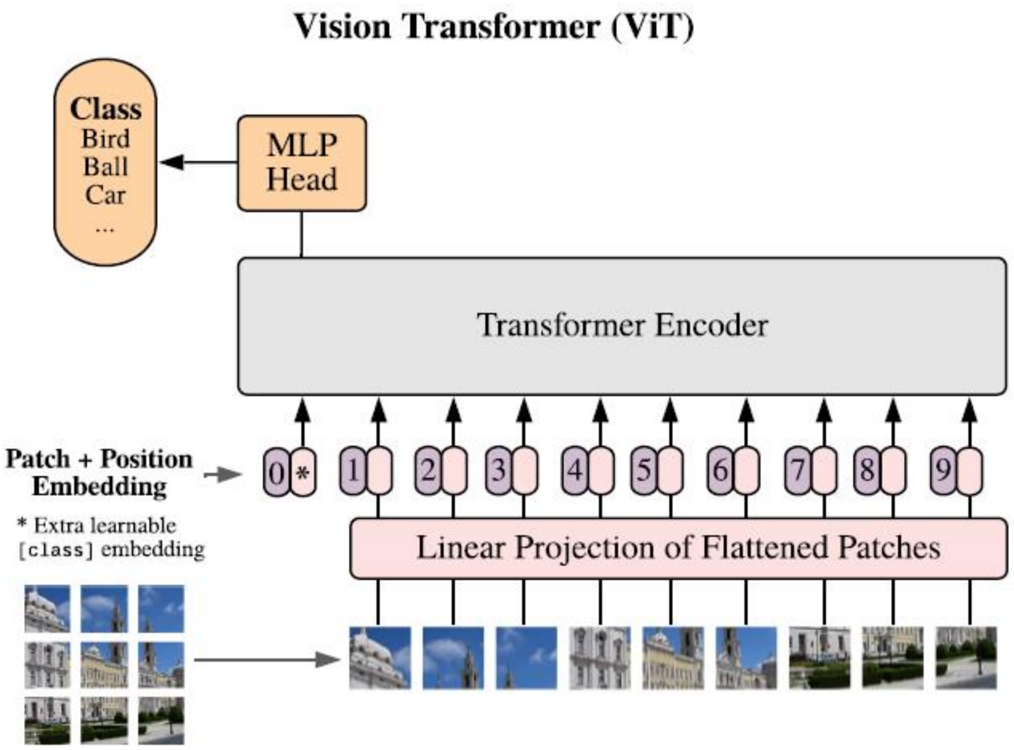
- Originally developed to replace RNNs in machine translation tasks (e.g., English-to-French).
- Mainly utilize MLPs and attention blocks.
- Attention blocks use the attention mechanism → matrix multiplication.



Typical Transformer architecture [VAS2017].

Transformers

- Evolved to analyze almost any type of inputs (text, images, video, multimodal data, etc.).
- Large Language Models (LLMs), for example ChatGPT, typically utilize Transformers.



Transformer for image analysis [DOS2020].

DNN training

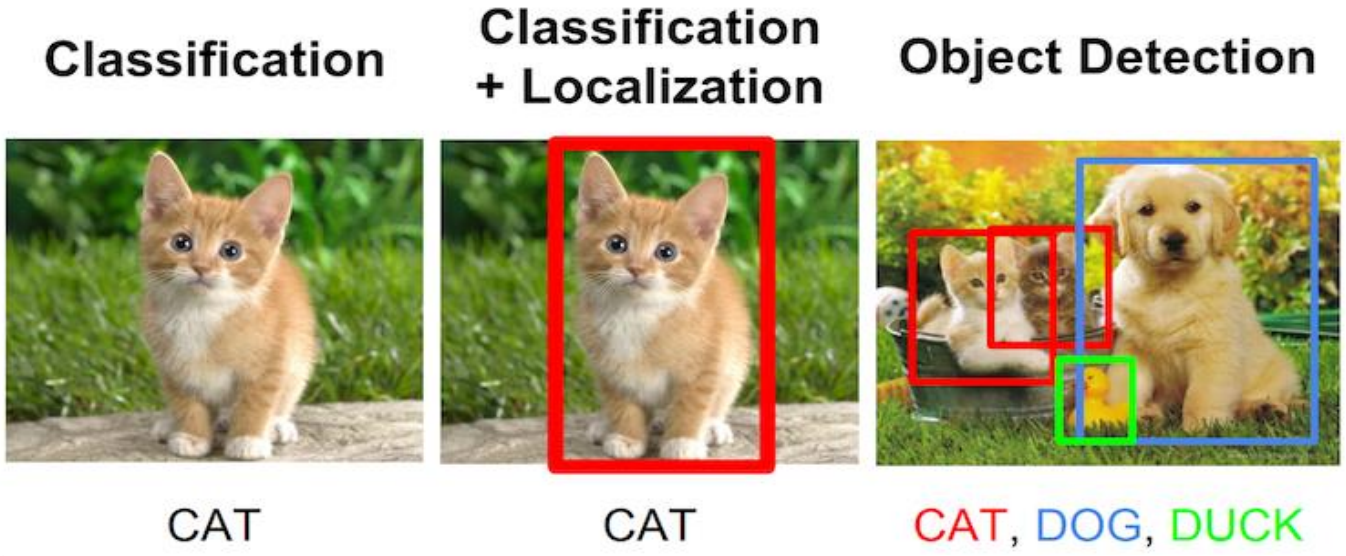
- All types of DNNs have trainable parameters.
- Trainable parameters are adjusted during training.
- Training:
  - Data (+ annotations).
  - Loss function (quantifies performance).
  - Optimizer (adjusts parameters based on loss function value).
  - Resources!

Contents

- Introduction
- Deep Neural Networks (DNNs)
- 2D object detection and tracking
- Semantic image segmentation
- Visual anomaly detection
- Human pose estimation
- Action/gesture recognition
- Applications

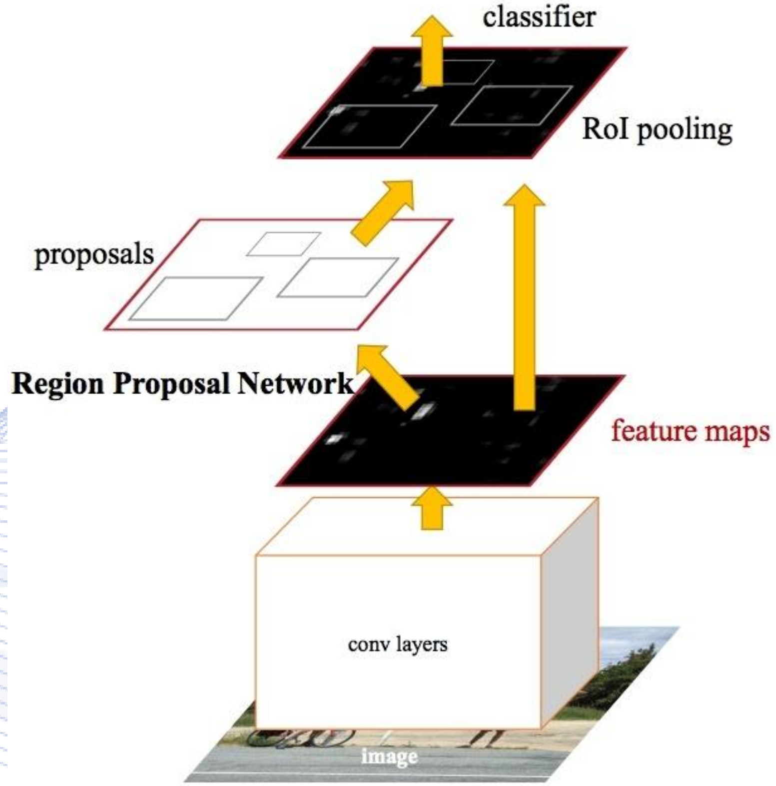
2D object detection

- 2D object detection: classification + 2D localization.
  - Find what is in an image and where it is.
  - Input: RGB image.
  - Output: 2D bounding boxes + class IDs.



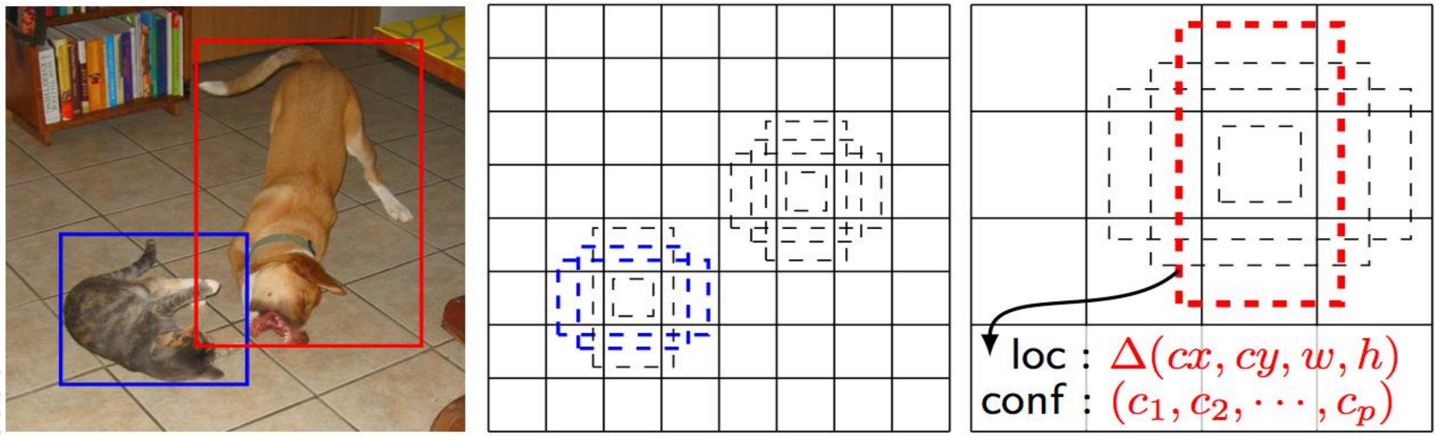
2D object detection

- Faster-RCNN [REN2015]: Utilizes a Region Proposal Network (RPN) to produce proposals based on a predicted objectness score.
- The proposals are extracted by a RoI pooling layer and are fed to an MLP for classification.
- Computation depends on the number of proposals.



2D object detection

- Single Shot Detector (SSD) [LIU2016]: Fully convolutional network that utilizes anchors and multiple resolution features.

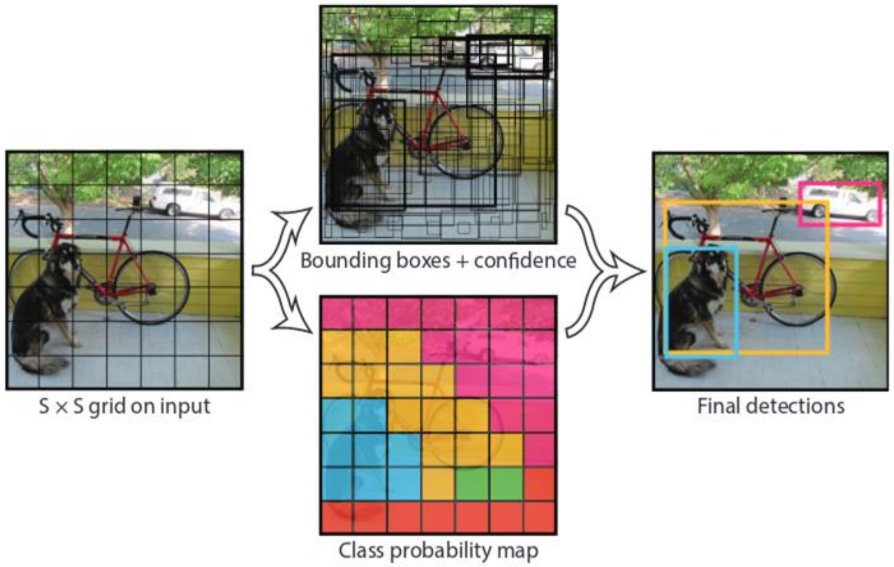


(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

- Example: The cat has 2 anchors matched in the 8×8 feature map, none matches the dog. In the 4×4 feature map one anchor matches the dog.

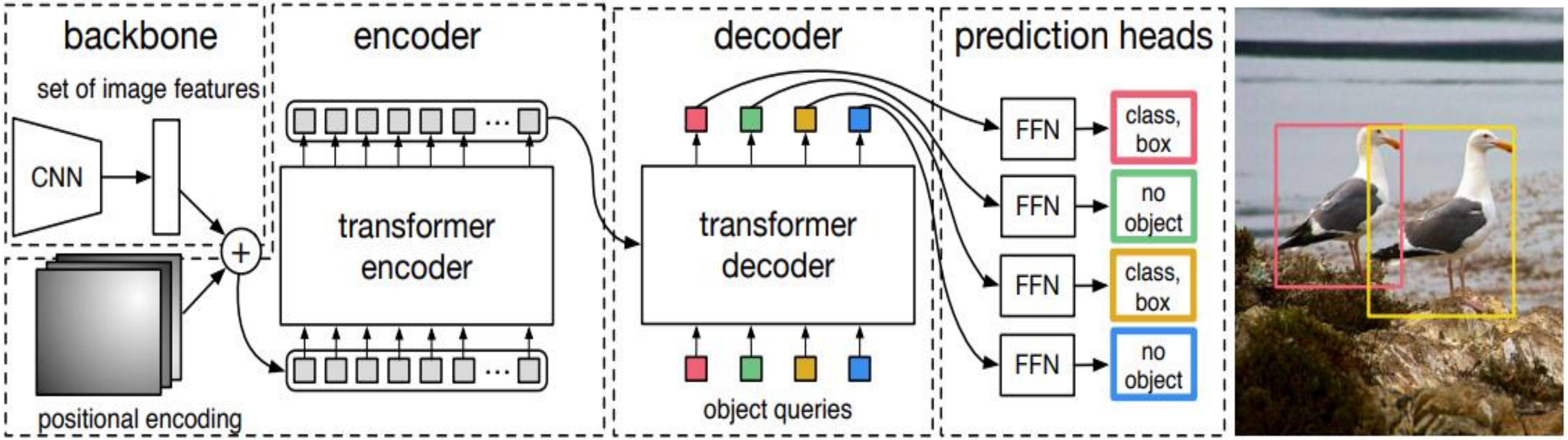
2D object detection

- YOLO [RED2016]: Divides input image into an $S \times S$ grid.
- For each grid cell, a class probability map is predicted.
- Also, using each grid cell as center, N bounding boxes are predicted along with the corresponding confidence scores.
- Final output is obtain using Non-Maximum Suppression (NMS).



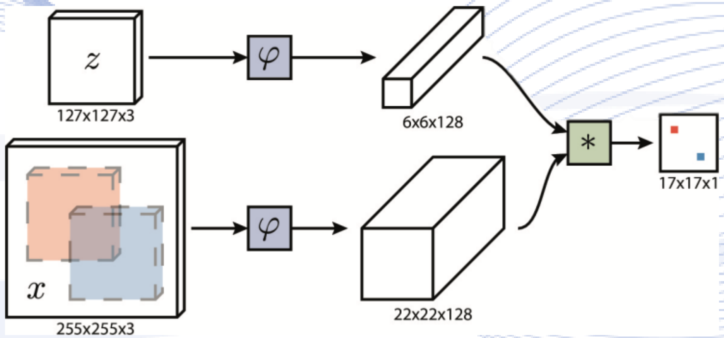
2D object detection

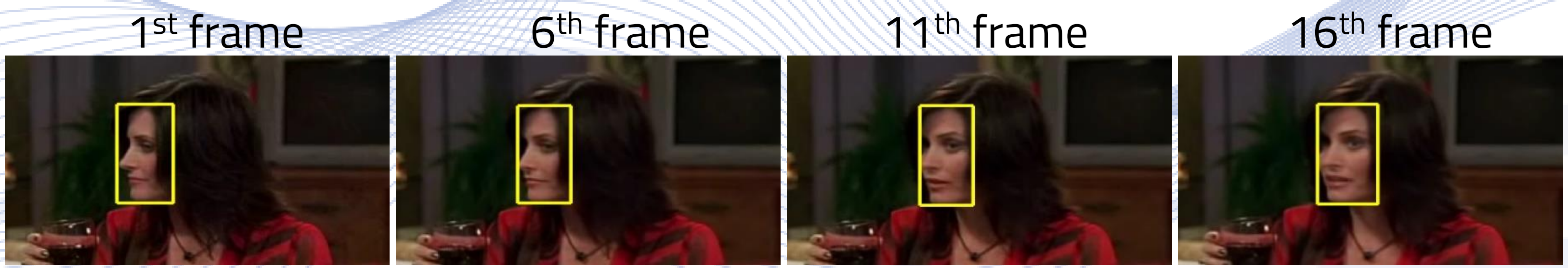
- DETR [CAR2020]: Utilize Transformers for 2D object detection.
  - No need for anchors or NMS algorithm.
  - Used on top of CNNs (features extracted by a CNN).



2D object tracking

- 2D object tracking: associates each detected bounding box in the current video frame with one in the next video frame.
  - SiamFC [BER2016]: CNN with 2D convolutional layers in Siamese configuration.



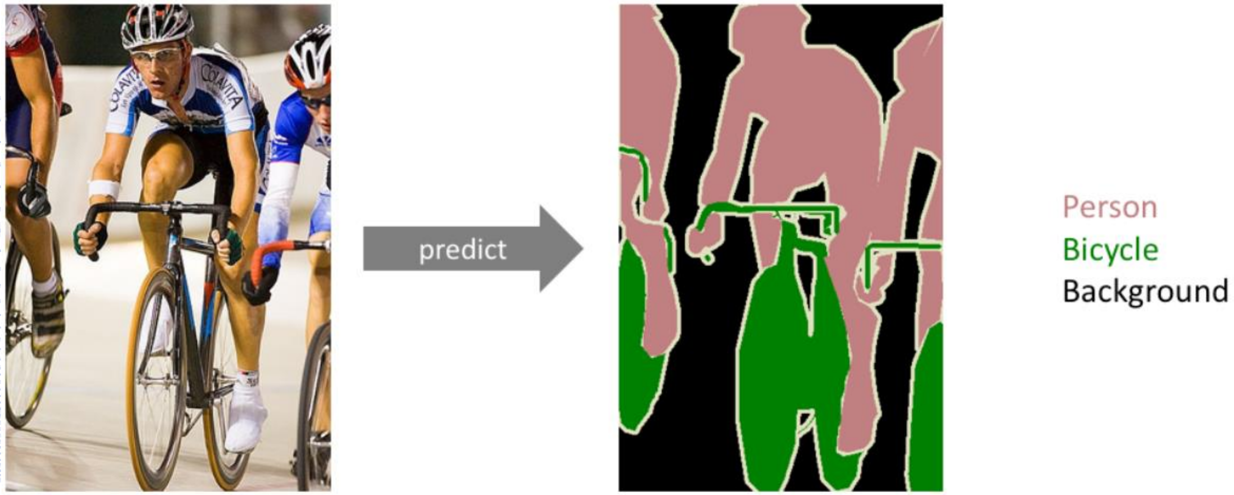


Contents

- Introduction
- Deep Neural Networks (DNNs)
- 2D object detection and tracking
- Semantic image segmentation
- Visual anomaly detection
- Human pose estimation
- Action/gesture recognition
- Applications

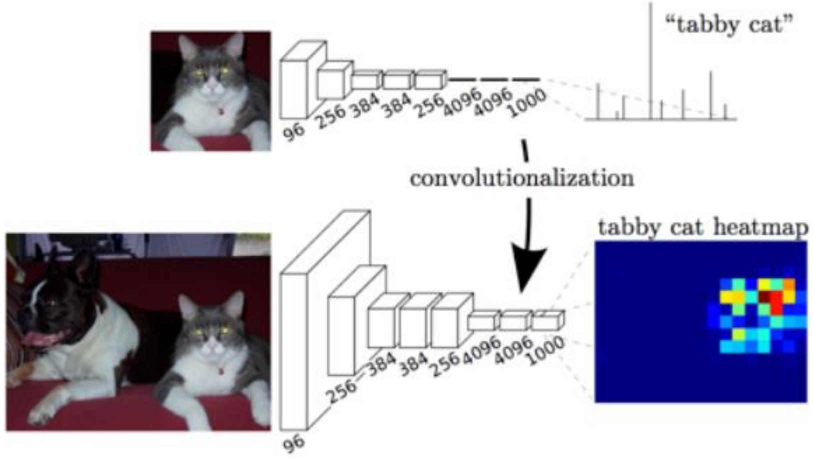
Semantic image segmentation

- Semantic image segmentation: classify each pixel of the input image to an object class.
  - Input: RGB image.
  - Output: 2D segmentation map.

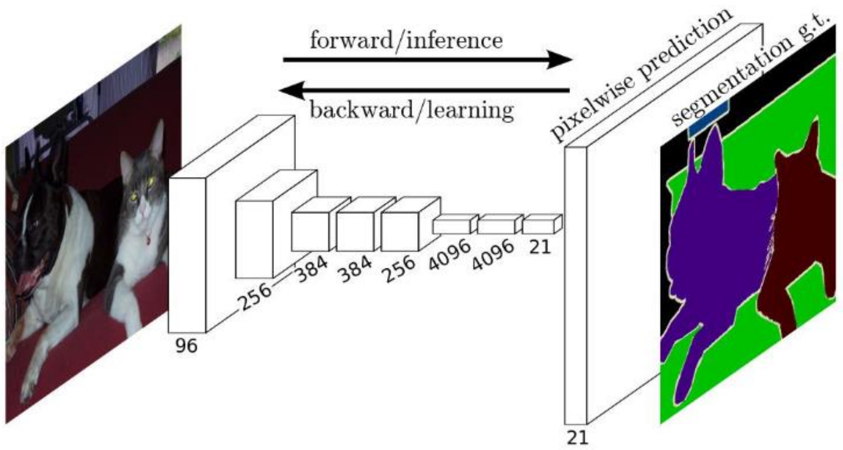


Semantic image segmentation

- Most simple approach: Replace final MLP layers of typical CNNs with convolutional ones.
  - Output class heatmaps.
- Add “decoding” convolutional layers → encoder-decoder.

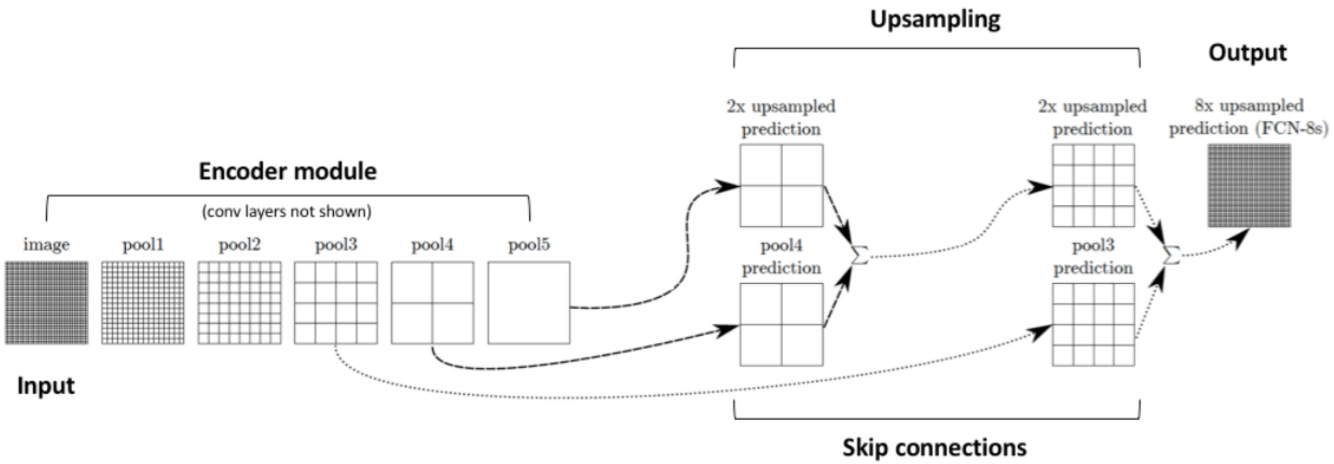


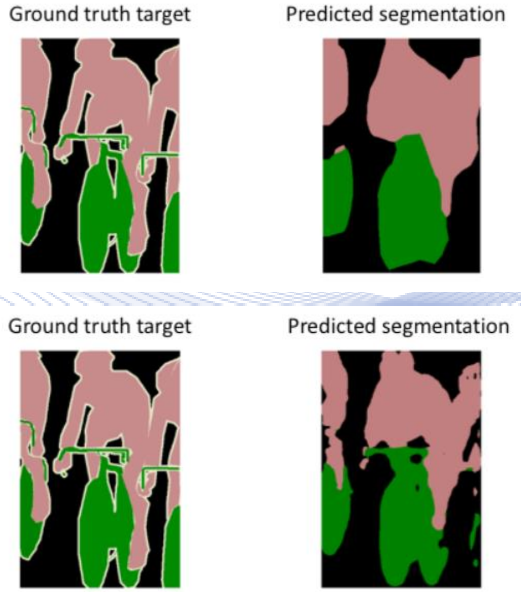
The encoder produces a ***coarse*** feature map which is then refined by the decoder module.



Semantic image segmentation

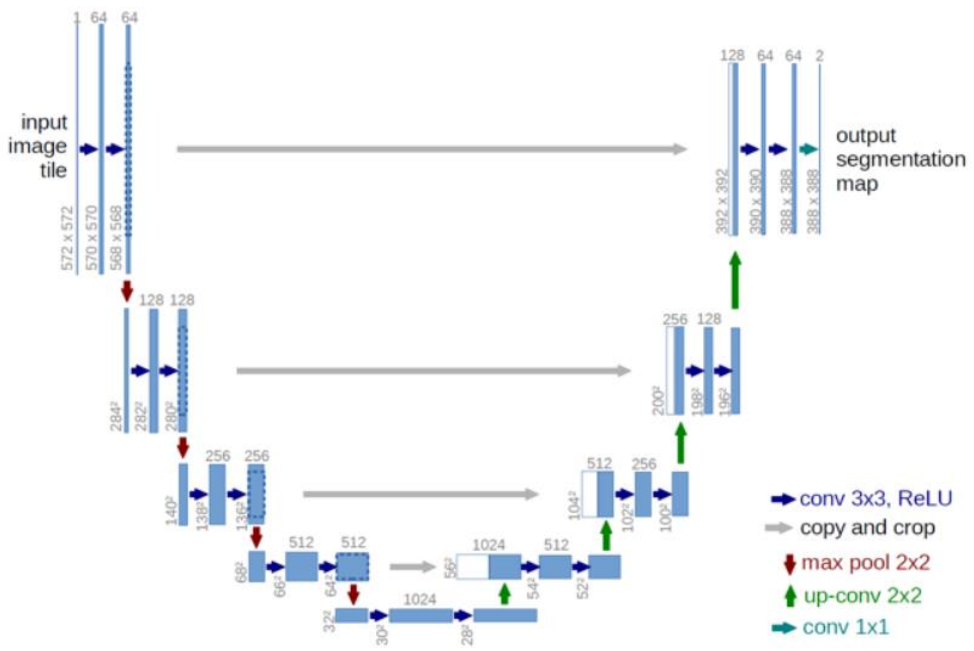
- Encoder radically reduces image resolution → coarse segmentation maps.
- Skip network connections between encoder and decoder.
  - Improved segmentation performance.





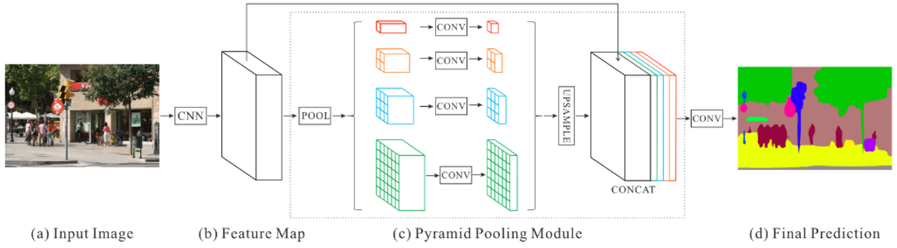
Semantic image segmentation

- U-Net [RON2015]: Symmetric encoder-decoder with skip connections.
  - Decoder capacity was expanded.
- Early features that preserve spatial information are enriched with semantic information → accurate results.
- Many variations: V-Net, U-Net++, ResUnet, U^2-Net, more.

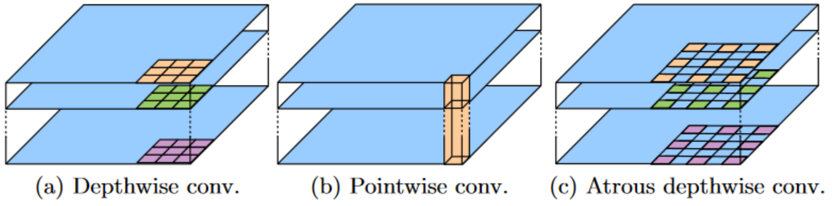


Semantic image segmentation

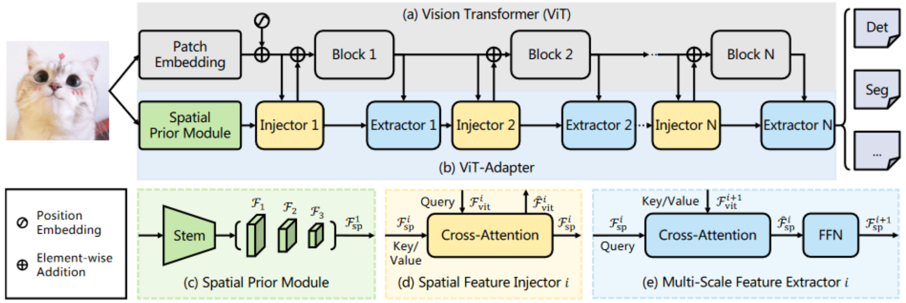
- Spatial Pyramid Pooling (SPP) [HE2015]:
  - Multi-scale features.
  - Can be slow.



- DeepLabV3+ [CHE2018]: Atrous Spatial Pyramid Pooling (ASPP) module.
  - Larger field of view, same computations.



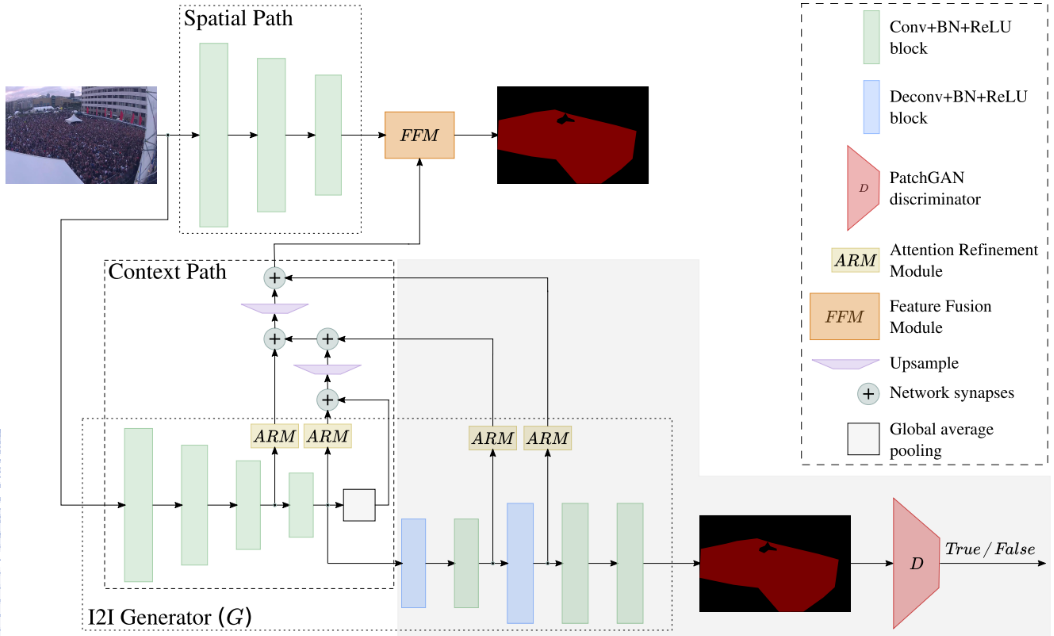
- ViT-Adapter [CHE2022]: Vision Transformer-based.
  - Huge number of trainable parameters (up to ~350M).



Artificial Intelligence & Information Analysis Lab

Semantic image segmentation

- I2I-CNN [PAP2021]: Real-time semantic image segmentation.
  - Complex architecture.
  - Goal: Remove "decoding" CNN.
- Utilizes Generative Adversarial Networks (GANs) and Image-to-Image Translation (I2I).
- Suitable for embedded execution.
  - Robots, UAVs, etc.

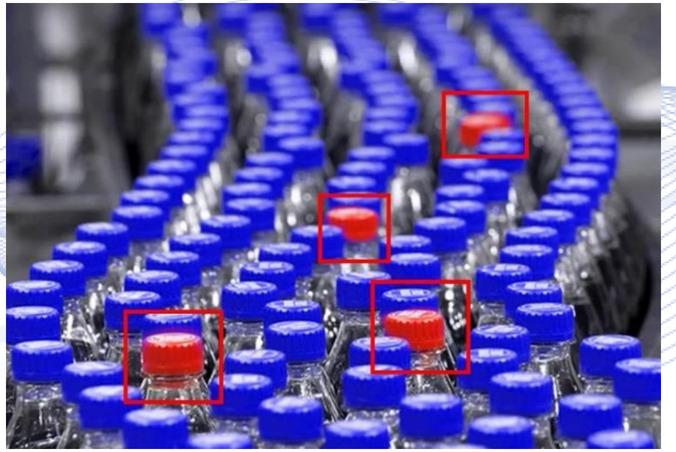


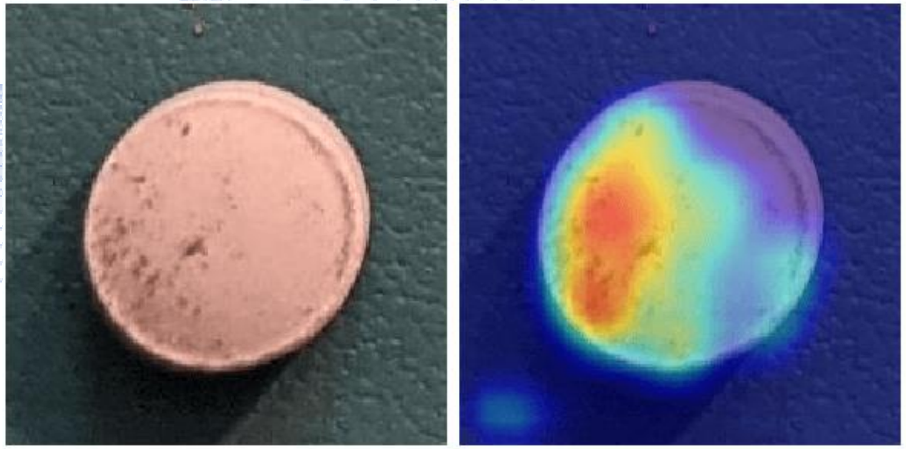
Contents

- Introduction
- Deep Neural Networks (DNNs)
- 2D object detection and tracking
- Semantic image segmentation
- Visual anomaly detection
- Human pose estimation
- Action/gesture recognition
- Applications

Visual anomaly detection

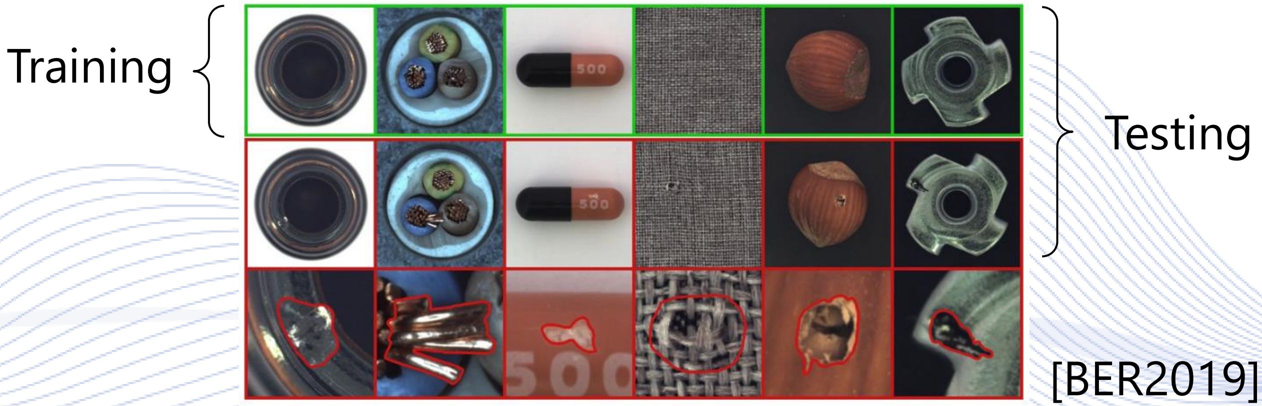
- Visual anomaly detection: identify unusual/unexpected patterns in the input image.
  - Identify (unknown) anomalies and optionally localize them.
  - Input: RGB image.
  - Output: Binary label (+ 2D bounding box/2D heatmap).

[NVDA]



[MATH]

Visual anomaly detection

- Training: Learn a DNN model using a large number of anomaly-free images only (+ artificial images with anomalies).
- Testing: Images with anomalies + anomaly-free images → detect deviations from learned model as anomalies.



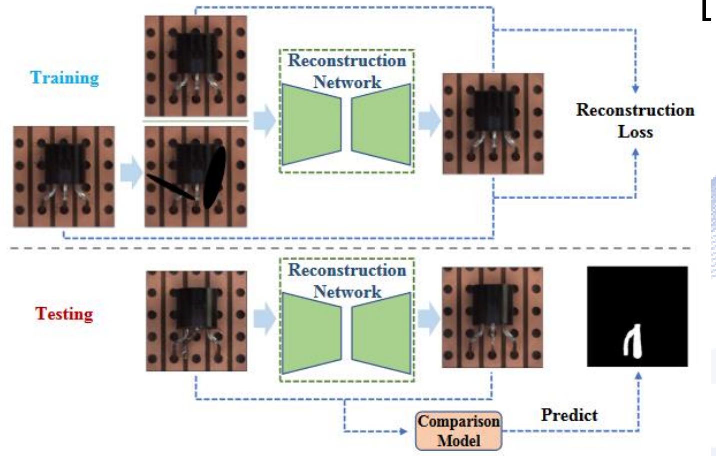
[BER2019].

- DNN types: CNNs, Autoencoders, Transformers.
- Excellent anomaly identification results in public datasets: >98%.

Visual anomaly detection

- Representation-based methods:
  - Rely on DNN extracted features.
  - Anomaly detection by measuring feature similarity.
- Reconstruction-based methods:
  - Learn to generate anomaly-free images.
  - Anomaly detection by comparing input image with generated anomaly-free image.

[LIU2023].

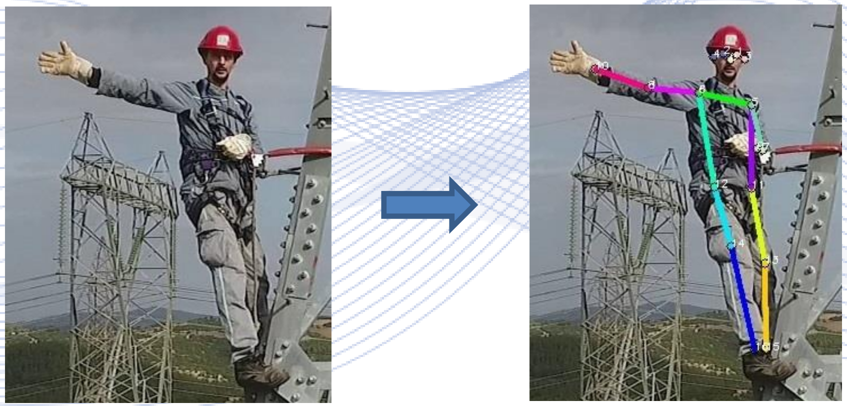


Contents

- Introduction
- Deep Neural Networks (DNNs)
- 2D object detection and tracking
- Semantic image segmentation
- Visual anomaly detection
- Human pose estimation
- Action/gesture recognition
- Applications

Human pose estimation

- Human pose estimation (HPE): Estimate the configuration of the human body parts.
  - Human body joint recognition and 2D/3D localization.
  - Input: RGB image.
  - Output: 2D/3D location of each human body joint.

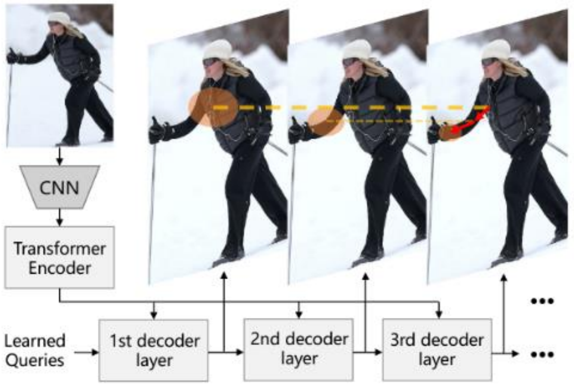


Human pose estimation

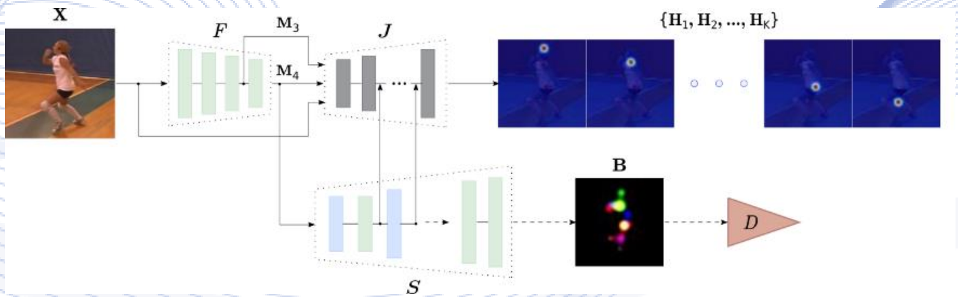
- Single-person HPE: Estimate pose of a single person that appears in an image/video.
- Multi-person HPE: Estimate pose of multiple persons.
  - Top-down approach: a) Detect each person. b) Estimate pose of each person.
  - Bottom-up approach: a) Detect all body joints. b) Grouping.

Human pose estimation

- 2D human pose estimation: Body joint locations in pixel coordinates.
  - Direct regression methods: Directly predict body joint locations.
    - Simple, lack accuracy.
  - Heatmap-based methods: a) Predict 2D body joint heatmaps. b) Obtain pixel coordinates by processing heatmaps.
    - Very accurate, heatmap resolution may affect accuracy.



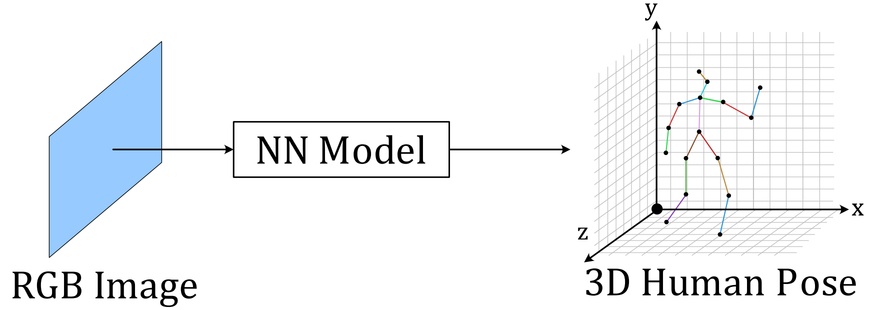
[LI2021].

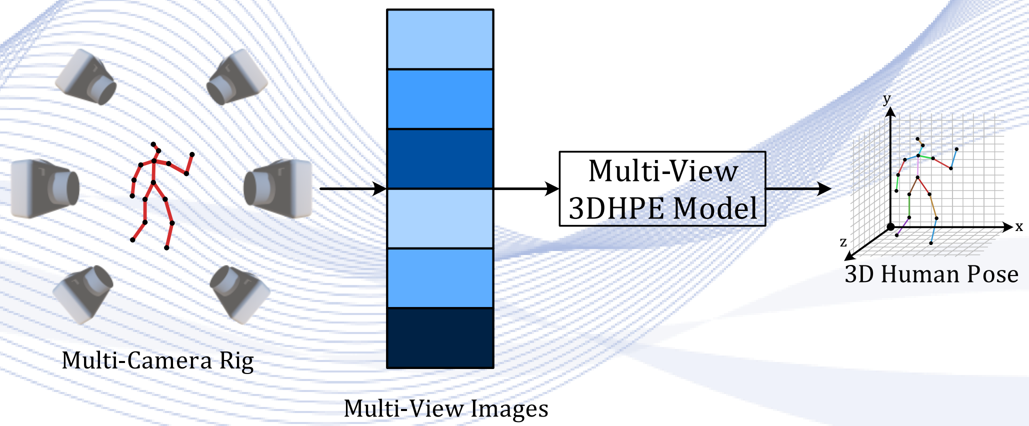


[PAP2022].

Human pose estimation

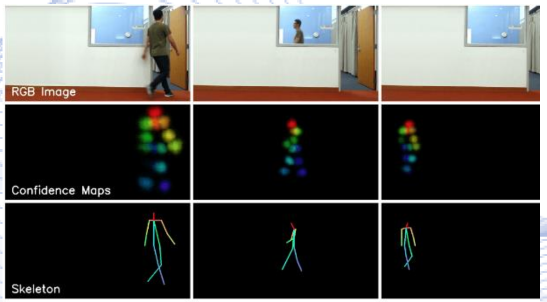
- 3D human pose estimation: Body joint locations in 3D world coordinates.
  - Monocular: Estimate human pose from single image/video.
    - Simple, lack accuracy.
  - Multi-view: Estimate human pose from multiple images/videos captured from different viewpoints.
    - Accurate, multi-view data are not easy to obtain.





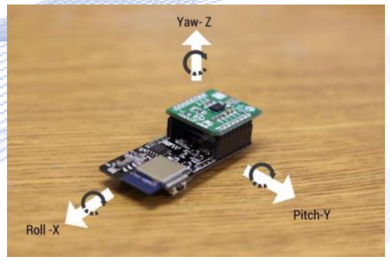
Human pose estimation

- DNN architectures:
  - Simple CNNs (direct regression).
  - Encoder-decoder CNNs (heatmap-based).
  - Transformers (direct regression, heatmap-based).
- Input sensors:
  - RGB cameras.
  - Depth sensors.
  - Inertial measurement units (IMUs).
  - Radio frequency devices.



[ZHA2018]


[TER]



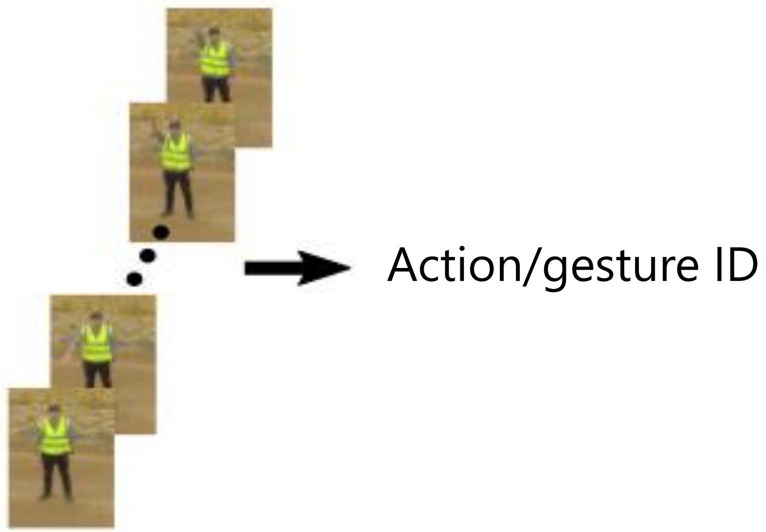
[HAC]

Contents

- Introduction
- Deep Neural Networks (DNNs)
- 2D object detection and tracking
- Semantic image segmentation
- Visual anomaly detection
- Human pose estimation
- Action/gesture recognition
- Applications

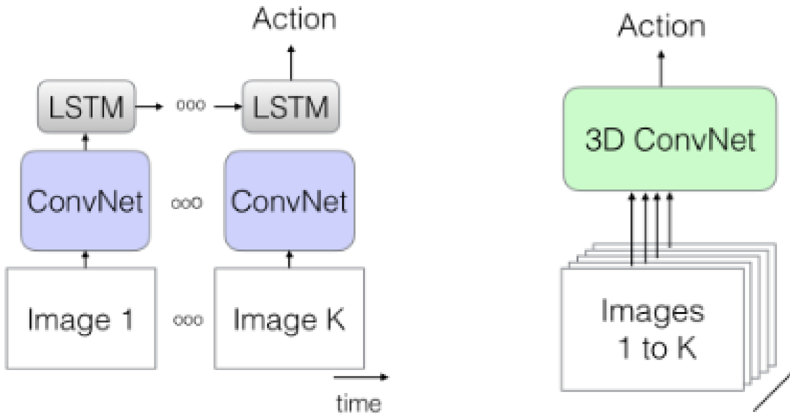
Action/gesture recognition

- Human action/gesture recognition: Identify the action/gesture performed by a human.
  - Input: RGB video.
  - Output: Action/gesture ID.

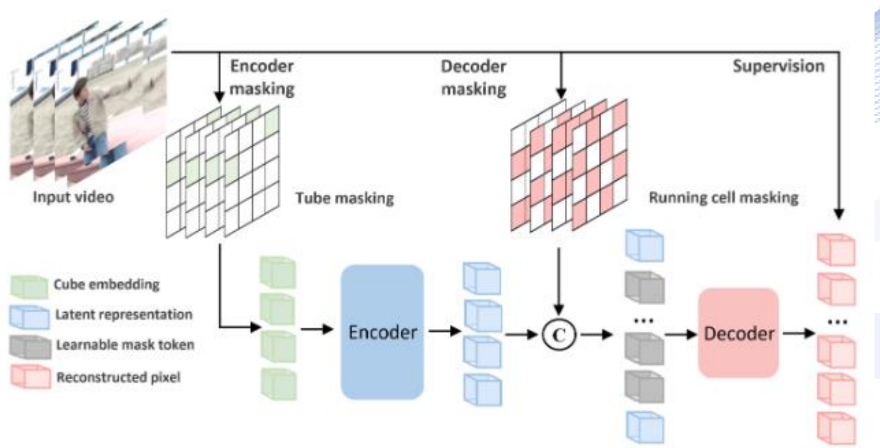


Action/gesture recognition

- LSTM-based:
  - Process input video with LSTMs.
- 3D CNN-based:
  - CNNs with 3D convolution layers.
  - Encode spatio-temporal information.
- Transformer-based:
  - Exploit powerful Transformer architectures for action/gesture recognition.
  - Effective training without labels (reconstruction).



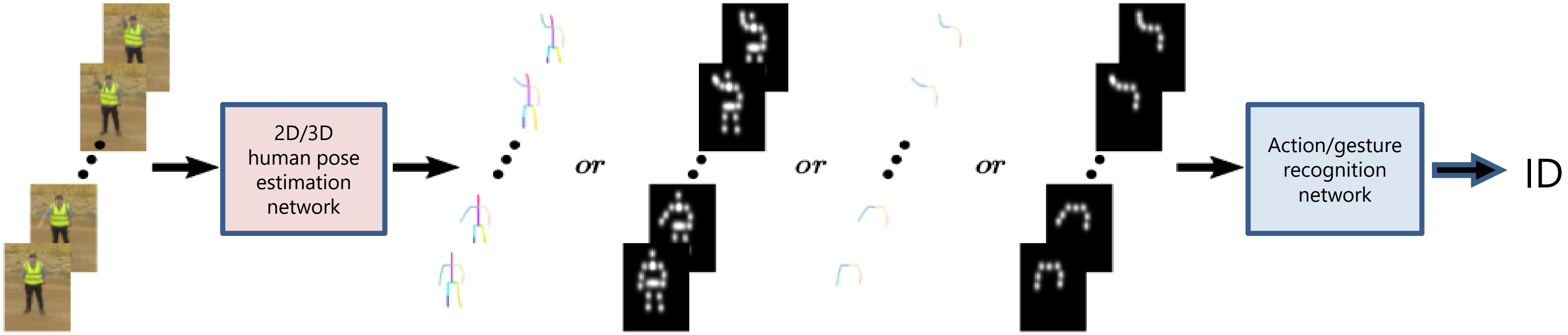
[CAR2017].



[WAN2023].

Action/gesture recognition

- Skeleton-based: Predict action/gesture ID by processing a sequence of 2D/3D skeletons → extracted using 2D/3D HPE.
  - Two-step approach.
  - Increased execution speed, high accuracy.
  - Action/gesture recognition DNNs: LSTMs, CNNs, Transformers, Graph Convolution Networks (GCNs).



[PAP2021b].

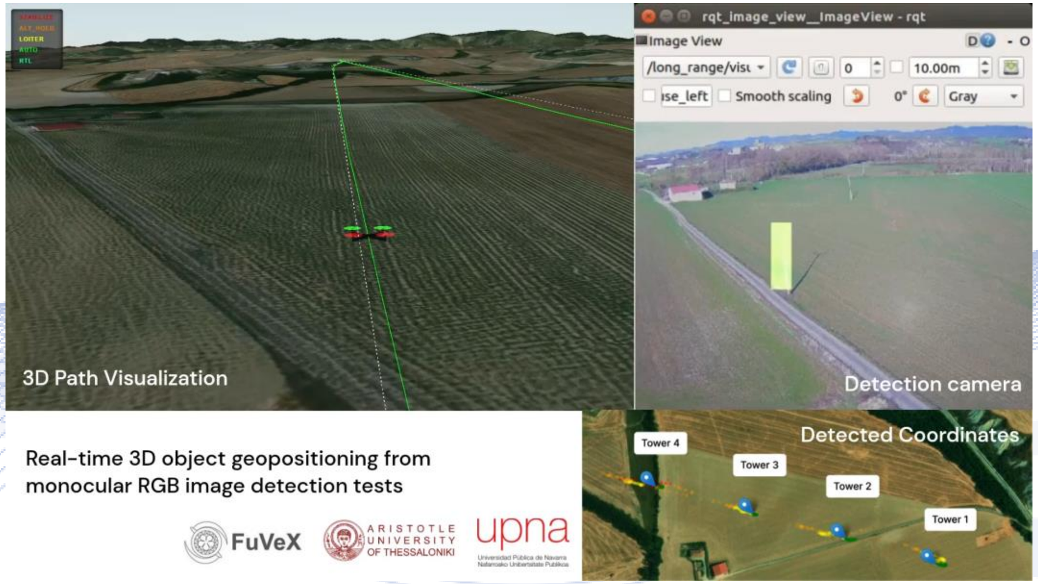
Contents

- Introduction
- Deep Neural Networks (DNNs)
- 2D object detection and tracking
- Semantic image segmentation
- Visual anomaly detection
- Human pose estimation
- Action/gesture recognition
- Applications

Applications

- Powerline elements detection and tracking.
  - Autonomous powerline elements inspection with UAVs.
  - 2D object detection + tracking.

[PAT2022].

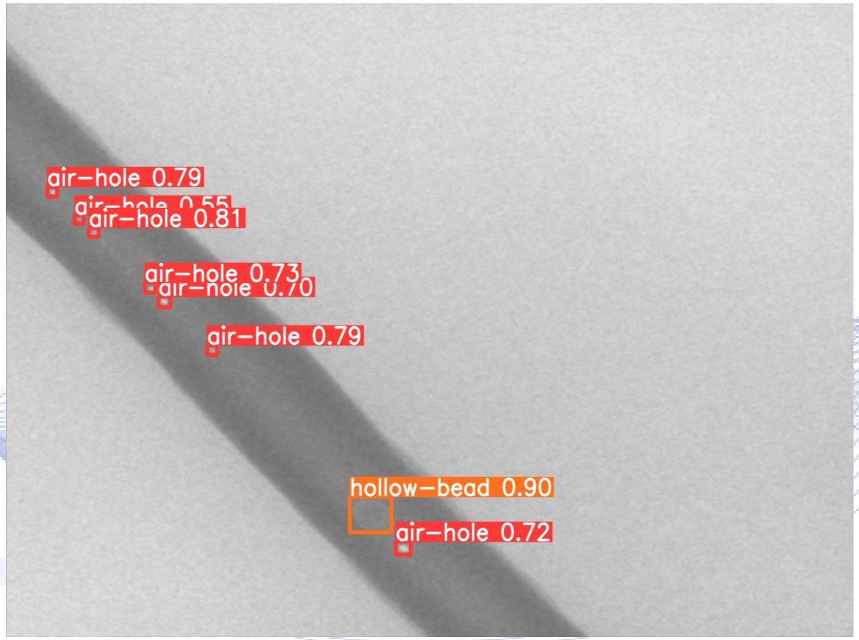


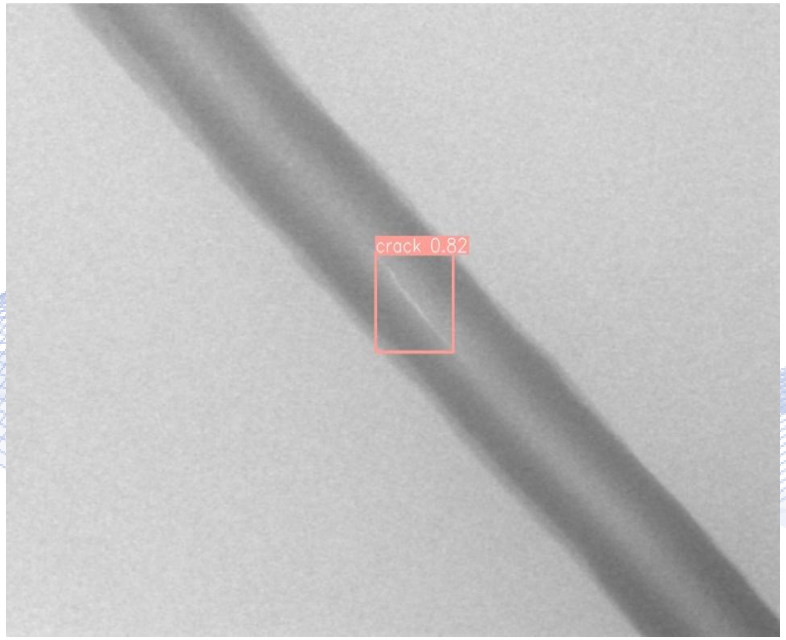
[ALA2023].

Artificial Intelligence & Information Analysis Lab

Applications

- Pipe damage detection in X-Ray images.
  - Autonomous pipeline inspection with UAVs.
  - 2D object detection + tracking.

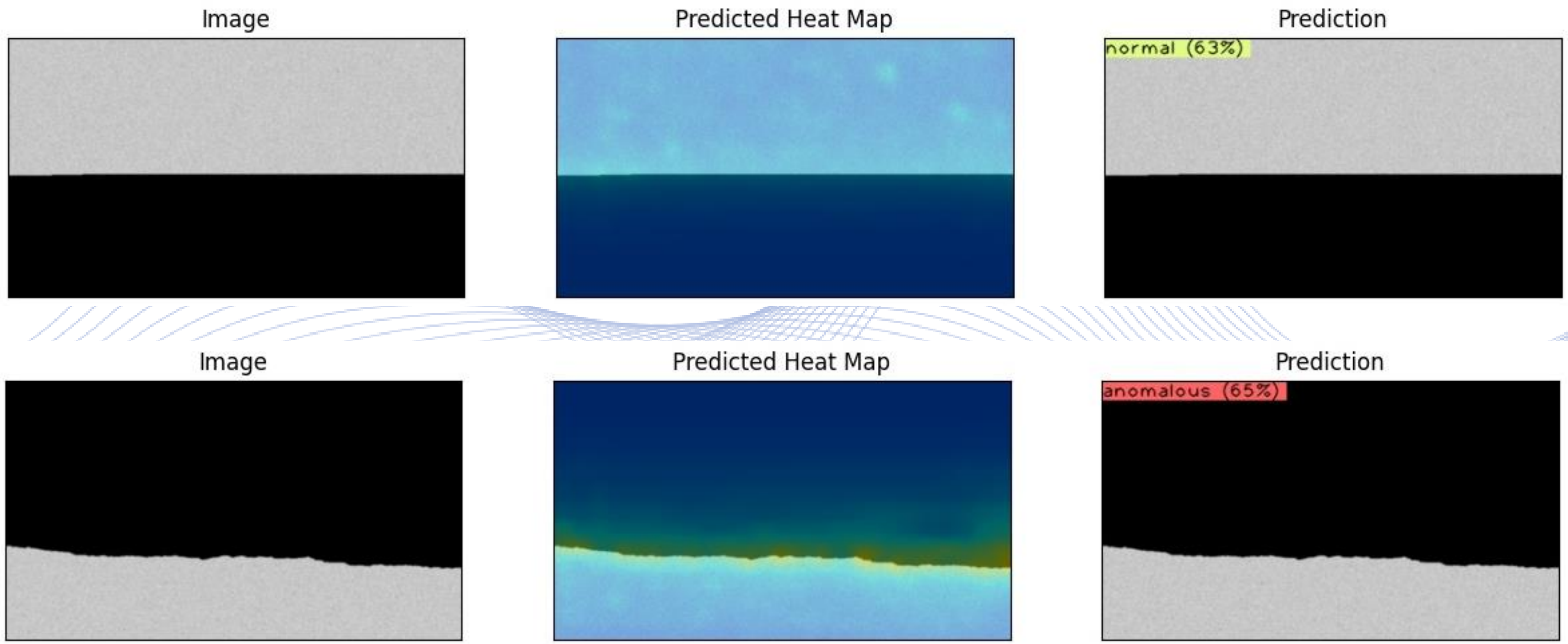




Artificial Intelligence &
Information Analysis Lab

Applications

- Pipe corrosion detection in X-Ray images.
  - Autonomous pipeline inspection with UAVs.
  - Anomaly detection.



Artificial Intelligence &
Information Analysis Lab

Applications

- Surrounding environment detection.
  - Autonomous powerline infrastructure inspection with UAVs.
  - 2D object detection + tracking + segmentation.

[PAP2022b].

Applications

- Pipe segmentation and damage detection.
  - Autonomous pipeline infrastructure inspection with UAVs.
  - 2D object detection + tracking + segmentation.

[PSA2024].

Applications

- Human crowd detection and avoidance.
  - Autonomous inspection with UAVs.
  - Image segmentation.

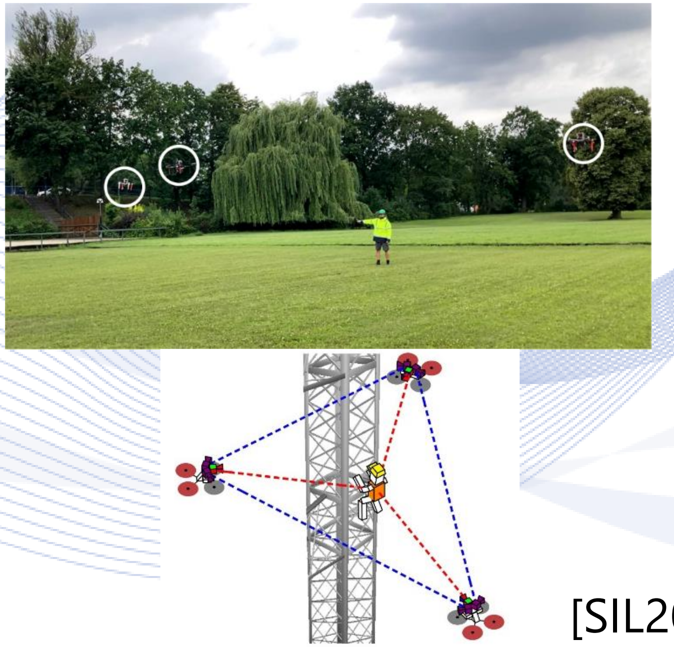
[PAP2021].

Applications

- Human worker state estimation.
  - Autonomous monitoring of human worker for safety.
  - Person detection + human pose/head pose estimation.

Applications

- Gesture recognition for human worker-UAV cooperation.
  - UAV formation control with gestures.
  - Person detection + human pose estimation + gesture recognition.



[SIL2023].

Applications

- Gesture recognition for human worker-UAV cooperation.
  - UAV control with gestures.
  - Person detection + human pose estimation + gesture recognition.

Q & A

Thank you very much for your attention!

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

References

[VAS2017] A. Vaswani, et al. "Attention is all you need." *Advances in neural information processing systems (NIPS)*, 2017.
[DOS2020] A. Dosovitskiy, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." *arXiv preprint arXiv:2010.11929*, 2020.
[REN2015] S. Ren, et al. "Faster R-CNN: Towards real-time object detection with region proposal networks." *Advances in neural information processing systems (NIPS)*, 2015.
[LIU2016] W. Liu, et al. "SSD: Single shot multibox detector." *Proceedings of European Conference on Computer Vision (ECCV)*, 2016.
[RED2016] J. Redmon, et al. "You only look once: Unified, real-time object detection." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
[CAR2020] N. Carion, et al. "End-to-end object detection with transformers." *Proceedings of European Conference on Computer Vision (ECCV)*, 2020.
[BER2016] L. Bertinetto, et al. "Fully-convolutional siamese networks for object tracking." *Proceedings of European Conference on Computer Vision Workshops (ECCVW)*, 2016.
[RON2015] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation." *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.
[HE2015] K. He, et al. "Spatial pyramid pooling in deep convolutional networks for visual recognition." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37, 9, 1904-1916, 2015.

References

[CHE2018] L.-C. Chen, et al. "Encoder-decoder with atrous separable convolution for semantic image segmentation." *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018.

[CHE2022] Z. Chen, et al. "Vision transformer adapter for dense predictions." *arXiv preprint arXiv:2205.08534*, 2022.

[PAP2021] C. Papaioannidis, I. Mademlis, and I. Pitas. "Autonomous UAV safety by visual human crowd detection using multi-task deep neural networks." Proceedings of the *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[BER2019] P. Bergmann, et al. "MVTec AD--A comprehensive real-world dataset for unsupervised anomaly detection." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[LIU2023] J. Liu, et al. "Deep industrial image anomaly detection: A survey." *arXiv preprints* arXiv:2301.11514, 2023.

[LI2021] Li, Ke, et al. "Pose recognition with cascade transformers." Proceedings of the *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[PAP2022] C. Papaioannidis, I. Mademlis, and I. Pitas. "Fast CNN-based single-person 2D human pose estimation for autonomous systems." *IEEE Transactions on Circuits and Systems for Video Technology*, 33, 3, 1262-1275, 2022.

[ZHA2018] M. Zhao, et al. "Through-wall human pose estimation using radio signals." Proceedings of the *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

References

[CAR2017] J. Carreira, and A. Zisserman. "Quo vadis, action recognition? a new model and the kinetics dataset." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2017.
[WAN2023] L. Wang, et al. "Videomae v2: Scaling video masked autoencoders with dual masking." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2023.
[PAP2021b] C. Papaioannidis, et al. "Learning fast and robust gesture recognition." *European Signal Processing Conference (EUSIPCO)*, 2021.
[PAT2022] E. Patsiouras, V. Mygdalis, and I. Pitas. "Whitening Transformation inspired Self-Attention for Powerline Element Detection." *Proceedings of the International Conference on Pattern Recognition (ICPR)*, 2022.
[ALA2023] D. Aláez, V. Mygdalis, J. Villadangos, and I. Pitas. "Real-time object geopositioning from monocular target detection/tracking for aerial cinematography." MMSP 2023. https://doi.org/10.5281/zenodo.8276584.
[PAP2022b] C. Papaioannidis, I. Mademlis, and I. Pitas. "Fast Semantic Image Segmentation for Autonomous Systems." *Proceedings of the IEEE International Conference on Image Processing (ICIP),* 2022.
[PSA2024] D. Psarras, C. Papaioannidis, V. Mygdalis, and I. Pitas, "A Unified DNN-Based System for Industrial Pipeline Segmentation", 2024.
[SIL2023] G. Silano, et al. "Human-Swarm Interaction with a Gesture-Controlled Aerial Robot Formation for Safety Monitoring Applications.", *IROS Workshop*, 2023.

References

[NVDA] https://developer.nvidia.com
[MATH] https://www.mathworks.com/help/images/detect-anomalies-using-single-class-classification.html
[TER] https://www.terabee.com
[HAC] https://www.hackster.io
[PIT2021] I. Pitas, “Computer vision”, Createspace/Amazon, in press.
[PIT2017] I. Pitas, “Digital video processing and analysis” , China Machine Press, 2017 (in Chinese).
[PIT2013] I. Pitas, “Digital Video and Television” , Createspace/Amazon, 2013.
[NIK2000] N. Nikolaidis and I. Pitas, “3D Image Processing Algorithms”, J. Wiley, 2000.
[PIT2000] I. Pitas, “Digital Image Processing Algorithms and Applications”, J. Wiley, 2000.