Exam ss20 - Exam ss20

Computer Vision III: Detection, Segmentation and Tracking (Technische Universität München)

Computer Vision Group
Department of Informatics
Technical University of Munich

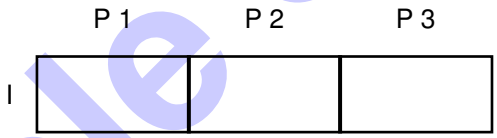
Esolution
Place student sticker here

Note:

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check list.

Computer Vision III: Detection, Segmentation, and Tracking

Exam:	IN2375 / Endterm	**Date:**	Friday 14th August, 2020
Examiner:	Prof. Leal-Taixe	**Time:**	08:00 – 09:30



Left room from _____ to _____

from _____ to _____

Early submission at _____

Notes _____

Computer Vision Group
Department of Informatics
Technical University of Munich

Endterm

Computer Vision III: Detection, Segmentation, and Tracking

Prof. Leal-Taixe
Computer Vision Group
Department of Informatics
Technical University of Munich

Friday 14th August, 2020
08:00 – 09:30

Working instructions

- This exam consists of **12 pages** with a total of **3 problems**.
  Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is 60 credits.
- Detaching pages from the exam is prohibited.
- Allowed resources:
- **Answers are only accepted if the solution approach is documented.** Give a reason for each answer unless explicitly stated otherwise in the respective subproblem.
- Do not write with red or green colors nor use pencils.
- Physically turn off all electronic devices, put them into your bag and close the bag.
- Multiple choice questions: Multiple correct answers per question are possible and every correct ticked/unticked box will reward you with 0.5 points.

Problem 1 Multiple Choice (12 credits)

Mark your answer clearly by a cross in the corresponding box. Multiple correct answers per question possible.

Mark correct answers with a cross ☒
To undo a cross, completely fill out the answer option ■
To re-mark an option, use a human-readable marking ✕■

a) Check all that apply for the Viola-Jones detector:

- [ ] The final classifier is a linear combination of all weak learners.
- [ ] It uses random forests to find the best weak learners.
- [ ] It follows a feature extraction + classification paradigm.
- [ ] It uses handcrafted Haar-like features.

b) Check all that apply for object proposals:

- [ ] Selective search is a method that outputs object proposals.
- [ ] YOLO detector uses object proposals.
- [ ] Fast R-CNN uses object proposals.
- [ ] They are boxes located at fixed intervals all over the image.

c) Check all that apply for object detectors:

- [ ] One-stage object detectors are faster but less accurate than two-stage object detectors.
- [ ] Faster R-CNN is a one-stage object detector.
- [ ] Fast R-CNN uses anchors.
- [ ] R-CNN does one forward pass through the backbone CNN for every proposal.

d) Which of the following statements is true about Message Passing Networks (check all that apply):

- [ ] They are invariant to node permutations.
- [ ] They can only encode pairwise interactions between node features.
- [ ] None of the statements is correct.
- [ ] They are linear functions of node and edge feature vectors.

e) Which of the following statements is true about PointNet (check all that apply):

- [ ] It generalises the concept of convolution to the domain of unstructured signals, such as point sets.
- [ ] It achieves invariance to rigid transformation by estimating canonical object pose via Principal Component Analysis, followed by rectifying point cloud based on the estimated transformation before learning the positional encodings.
- [ ] None of the statements is correct.
- [ ] It achieves permutation invariance by concatenating all point coordinates and applying a multi-layer perceptron, followed by permutation-invariant max-pool operation.

f) Check all that apply for DeepLabv3+:

- [x] It gives a semantic segmentation map as output.
- [ ] It uses of depth-wise separable convolutions.
- [ ] It uses an attention mechanism on the input image in order to predict the scale at which each region of the image should be decoded.
- [x] It use of dilated convolutions.

Problem 2 **Short questions (26 credits)**

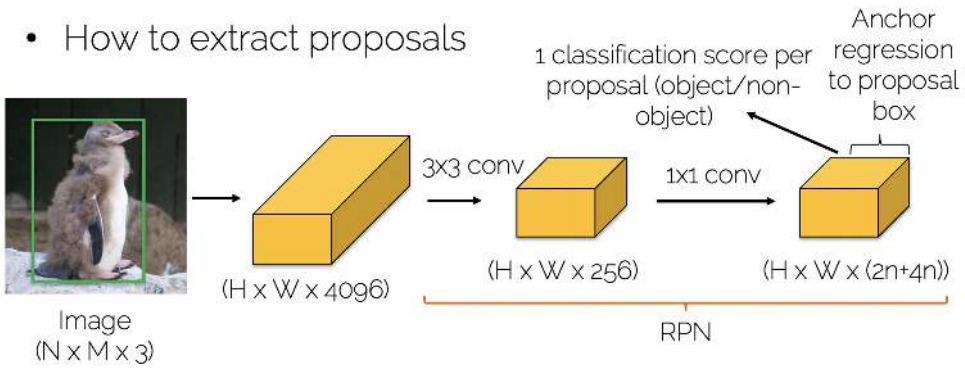
0 1 2 a) Name two problems of traditional object detection methods that follow a template matching plus sliding window approach.

(any two)
1. Occlusions (we need to see whole/most of the object). 2. This works to detect an instance of an object but not a class of objects. 3. Objects have an unknown position, scale and aspect ratio, the search space is searched inefficiently with sliding window

0 1 2 b) How is bounding box overlap measured in the Non-Maximum suppression (NMS) algorithm (1p)? Write the equation for such an overlap measure. What is the problem of using NMS on an object detector's output for heavily crowded scenes where we are interested in performing pedestrian detection (1p)?

Intersection over union or Jaccard index. (1p)
In crowded scenes, we expect pedestrians boxes to overlap a lot. If the threshold for NMS is too low, then even bounding boxes that contain pedestrians will be erased. (1p)

0 1 2 c) You are using Faster R-CNN, and your region proposal network takes as input a feature map of size $H \times W \times 2056$. What type of operations do you use to have the desired output? Clearly draw your architecture (1p). Express the output in terms of the number of anchors n (1p).



d) ExtremeNet (X. Zhou et al. “Bottom-up object detection by grouping extreme and center points”. CVPR 2019) is a network that outputs 4 heatmaps for the extreme corners of an object plus a heatmap for the center of the object. Name the main reason behind predicting extreme points as opposed to bounding box corners (1p)? How does ExtremeNet use the center heatmap to estimate the final bounding boxes (1p)?

0 1 2

Reason: Predicting bounding box corners can be hard if the corner does not overlap with the object, but is rather in the background. The network is then not focused on learning the object shape and finding its boundaries, but rather in learning an "artificial" representation of an object. (1p)

ExtremeNet uses a voting system. All corners (extremes) of an object vote for a center position. If this center position matches a center found in the heatmap, then the object is accepted. (1p)

e) You want to perform human pose estimation with neural networks. The image contains only one person. Name two ways to express the output so that you can recover the skeleton at test time.

0 1 2

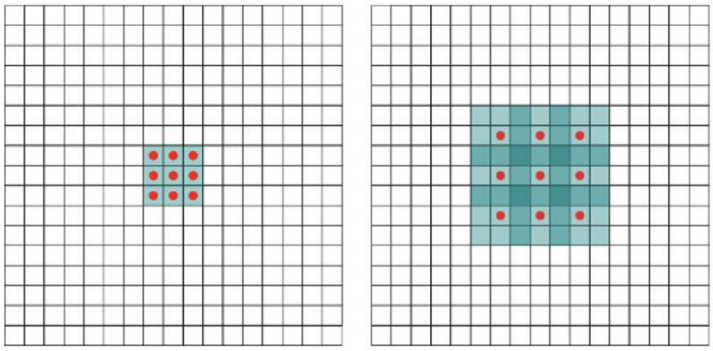
1. Direct regression, where the body joint positions are expressed as image pixels. 2. Heatmap prediction, you predict a full image per joint, this expresses the likelihood of each pixel showing that specific joint.

f) What are dilated (atrous) convolutions (1p)? Draw an example (1p). Why are they useful in the context of semantic segmentation (1p)?

0 1 2 3

1. Convolutions where the weights are spread out (dilated), so that we can increase the receptive field of the convolution without increasing the number of weights or parameters. (1p)

Drawing (1p). Note, the filter is marked by the red dots, the green area shows the receptive field. The initial kernel in this example is 3x3, while the dilated one is 5x5.



3. They are useful to work with higher resolution inputs without increasing the number of parameters needed, so that we do not lose information during the pooling operations. (1p)

(the fact that the number of parameters does not increase has to be mentioned for full points).

0 1 2

g) Explain how (vanilla) Tracktor performs data association for multiple object tracking (1p). How does Tracktor account for new objects entering the scene in the middle of the video (1p)?

You start by detecting all boxes in an image, you then feed those boxes as proposals for the next time step and you regress those boxes. If you feed in a box as a proposal, you give the same ID to the location where the box has been regressed. (1p)
New objects entering the scene are detected using the original detector RPN, and performing Non-maximum suppression between the previous bounding boxes and the new detections to find out if a new target has appeared on the scene. (1p) (-0.5 if NMS is not mentioned)

0 1 2 3

h) Explain the 3 training steps of OSVOS (S. Caelles et al. "One-shot video object segmentation". CVPR 2017), detailing the purpose of each step (1p per step).

1. Pre-training: the network is pre-trained on ImageNet for the task of object classification. (1p)
2. Parent network training: the network is trained for the general task of video object segmentation with all the training set sequences. (1p)
3. Fine-tuning: the network is fed with the first frame ground truth mask of the object. It is then fine-tuned on this mask (plus data agumentation of the mask), in order to learn its appearance and to know *which* object to segment in the next frames. (1p).

0 1 2

i) How many multiplications are done in a layer of depth-wise separable convolutions with 5*x*5 kernels on a feature map of 9*x*9*x*7 (no padding, stride of 1). There is no need to solve the multiplication, just write down the operations (multipliers) (1p) and their meaning (1p).

7 kernels of size 5x5x1 applied to 5x5 locations = (7x5x5) x (5x5) locations (1p for expressing the right multiplication, 1p for the meaning of the variables).

0 1 2

j) Explain how *stuff* and *things* are treated differently in the panoptic head of UPSNet (Xiong et al., "UPSNet: A Unified Panoptic Segmentation Network". CVPR 2019).

Stuff logits are passed directly to the panoptic logit map. (1p)
Things logits need to be first masked by the corresponding instances. (1p)

k) Explain the major problem of training a deterministic (one-to-one mapping) Deep Learning Model (e.g. S-LSTM) with an $L2$ loss for pedestrian trajectory prediction (1p). How could you extend the model to solve this problem (1p)?

0 1 2

Problem of trajectory prediction is multimodal/ stochastic (1p) - Optimisation of L2 loss results in unrealistic average path/ linear trajectory
Using generative (GAN, VAE) or stochastic model (e.g. Gaussian mixture model) (1p).

l) Name one advantage (1p) and one disadvantage (1p) of using voxelized representation of a 3D point cloud for representation learning?

0 1 2

+ We can simply apply 3D convolutions on this voxelised representation, thus leverage existing convolutional architectures and pre-trained networks - This approach is memory consuming (course of dimensionality: number of cells grows exponentially with respect to the grid dimension) - Early quantisation of the signal and loss of detail

Problem 3 Long question (22 credits)

You are building part of the vision pipeline that should be built into a car to provide it with autonomous driving capabilities, and your task is to perform multi-object tracking of *cars* and *pedestrians*, for which you have enough training data. For now you are building a prototype, so you are not concerned with computational time.

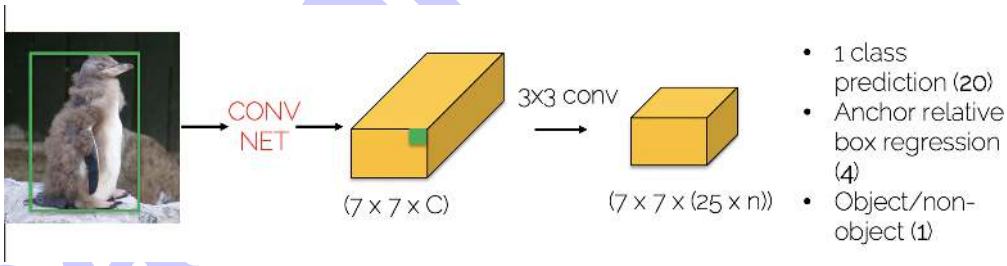
0 1 a) You decide to follow a tracking-by-detection paradigm. Explain the steps of such a methodology for tracking.

In tracking-by-detection, the multi-object tracking problem is divided into two steps: (i) first an object detector is applied to each frame of the image; (ii) a data association step links detections belonging to the same trajectory across frames.

0 1 b) What is the key difference between the tracking-by-detection and the tracking-before-detection paradigm?

Tracking before detection aims to initialise tracks based on category-agnostic bounding box representation of the input signals. This is based on generic objectness cues, such as spatial proximity, motion consistency and appearance similarity. At this stage, we are not know to which semantic classes our tracks belong. In contrast, in case of tracking-by-detection, we only track objects that we have already recognised (classified).
(Stating we first track the objects with a motion model and then "detect" them does not give points, this is pretty much just expanding the wording "tracking-before-detection". The notion that the class is not know, therefore, we need to run the classification after tracking needs to be stated to get the points).

0 1 2 3 4 c) To perform object detection, you implement a version of YOLOv2 (Redmon and Farhadi, "YOLO9000: Better, faster, stronger", CVPR 2017). Draw its pipeline (2p) with all relevant operations, specify the dimensions of its output when we have $C = 2$ semantic classes and n anchors (1p), and what losses are used to train the detector (1p).



(The key of the architecture is the size of the feature map before the 3x3 conv and the fact that it is a convolution that brings the output representation to its final size. (1x1 conv is also accepted)).

For $C = 2$ semantic classes, we will have $7x7x(4 + 2 + 1)xn = 7x7x7xn$ (1p)

(Accepted also if 2 values were used for the object/non-object classification).

Losses (1p if all correct):

1. Regression loss for the relative anchor box regression (4 values): L1 or L2 loss both accepted.
2. Classification loss for the C semantic classes. (C values): Cross-entropy loss
3. Classification loss for object/non-object. (1 value): BCE loss. (Can be put together with loss in 2.)

d) One-stage object detectors like YOLOv2 suffer from a particular problem when training, which makes them less accurate than two-stage detectors. What is that problem (1p)? Propose a solution by just changing the loss function (provide the name for the new loss function and the formula - 2p)?

0 1 2 3

Problem with one-stage detectors: They do the object/non-object classification plus the semantic classification at the same time. This means, all locations (n anchors per location) have to be analyzed. Most of these locations contain n useful information and only a handful have useful signal for training. To summarize, there is a heavy background/foreground imbalance. (1p)

Solution:
Focal loss (as in RetinaNet) (1p).
Formula: $FL(p_t) = -(1-p_t)^{\gamma}\log(p_t)$. Note when $\gamma = 0$ it is equivalent to cross-entropy loss. (1p)

Your code is now trained to detect cars and pedestrians, so you move to the temporal domain. You now want to track these objects, so you decide to use the network flow formulation for multiple object tracking (MOT). In that formulation, MOT is mapped into a graph.

e) What do nodes and edges represent in that formulation (1p)? What does it mean if an edge is *active* (1p)?

0 1 2

Nodes represent object detections, edges represent possible connections between detections. (1p)
An active edge means two nodes (detections) are connected, thereby forming a trajectory (1p).

f) Write down the objective function for the MOT problem, with cost variables $C(i,j)$, $C_{det}(i)$, $C_{in}(i)$, $C_{out}(i)$ and corresponding flow indicator variables $f(i,j)$, $f_{det}(i)$, $f_{in}(i)$, $f_{out}(i)$ (1p)?

0 1

$$\mathcal{T}* = \arg\min_{\mathcal{T}} \sum_i \mathrm{C_{in}(i)f_{in}(i)} + \sum_{i,j} \mathrm{C_t(i,j)f_t(i,j)} + \sum_i \mathrm{C_{det}(i)f_{det}(i)} + \sum_i \mathrm{C_{out}(i)f_{out}(i)}$$

(Not writing explicitly the minimization does not gives points).

0 1 2 3

g) You first use a mapping between 2D distances and cost $C(i, j)$, but you quickly realize this is not powerful enough, especially in crowded scenes. You decide to use more powerful appearance models based on a re-identification network trained for similarity learning. What architecture (with a ResNet-50 backbone) (1p) and loss (1p) do you use? Write down the formula for that loss (1p).

Architecture: siamese network (1p). Explaining the shared weights also gives a point even if the name siamese is not mentioned.
Loss: Triplet loss (1p, contrastive loss is also accepted)
Formula: $\mathcal{L}(A, P, N) = \max(0, |||f(A) - f(P)||||^2 - |||f(A) - f(N)||||^2 + m)$, where A indicates the anchor image, P the positive image, N the negative image and m the margin. (1p)

The formula needs to be in the form of the loss function, not the equation that leads to the loss function.

0 1 2 3

h) What is the main evaluation metric for multiple object tracking (1p)? State its formula (1p) and what each of the terms means (1p)?

MOTA = multiple object tracking accuracy. (1p)

- FP = False positives
- FN = False negatives (missing detections)
- IDsw: identity switches

Multi-object tracking accuracy → $\text{MOTA} = 1 - \frac{\sum_t (\text{FN}_t + \text{FP}_t + \text{IDSW}_t)}{\sum_t \text{GT}_t}$, ← Ground truth

0 1 2 3 4

i) Even if you have access to only video frames, you want to develop a 3D multi-object tracking method. Note, you do not have access to LiDAR data. Propose an approach with which we can leverage the components discussed in the lecture without modifications. Argue why your choices are the best.

1. Predict the depth of the scene (e.g., with a monocular depth neural network) 2. Convert depth maps to point clouds, treat them as a LiDAR signal 3. Use existing 3D object detectors on this representation (re-training will be required though). PointRCNN would be ok, but possibly we would like to use detector that also uses image data, not only relying on point clouds. 4. For tracking, I would not recommend to go with AB3DMOT because it is relying on precise 3D localization too much. GNN3DMOT would be a better approach, as it leverages both images and ?3D signal?, where 3D signal in this case will be point cloud representation of the estimated depth map.

Additional space for solutions–clearly mark the (sub)problem your answers are related to and strike out invalid solutions.