



## CV3 2022 Solution

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



Scan to open on Studocu

**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 3: Detection, Segmentation, and Tracking

**Exam:** IN2375 / Endterm  
**Examiner:** Prof. Dr. Laura Leal-Taixe

**Date:** Monday 28<sup>th</sup> February, 2022  
**Time:** 12:00 – 13:00

### Working instructions

- This exam consists of **12 pages** with a total of **2 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:
  - one **non-programmable pocket calculator**
  - one **analog dictionary** English ↔ native language
- Subproblems marked by \* can be solved without results of previous subproblems.
- **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.

## Problem 1 Multiple Choice (12 credits)

Mark your answer clearly by a cross in the corresponding box. Multiple correct answers per question possible. For every question, you will either get full credit (if you mark all the correct answers, and not mark all the incorrect answers) or no credit otherwise.

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) Which of the following is true for image segmentation (check all that apply):

- ☐ Decoders usually use pooling layers.
- ☐ Decoders usually use recurrent layers.
- ☐ Decoders usually use convolutional layers.
- ☐ Decoders usually use conditional random fields.

b) Which of the following is true for optical flow (check all that apply):

- ☐ FlowNet fuses the information using a convolutional or correlation layer.
- ☐ FlowNet can be used with a single image.
- ☐ Optical flow shows the real motion of the object.
- ☐ Optical flow shows the perceived 2D motion of the object.

c) Which of the following is true for object detection (check all that apply):

- ☐ YOLO is an one-stage detector, as is Faster R-CNN.
- ☐ DETR uses positional encoding.
- ☐ Fast R-CNN does one forward pass through the backbone CNN for every proposal.
- ☐ Fast R-CNN uses anchors.

d) Which of the following is true for metric learning (check all that apply):

- ☐ Metric learning is the task of finding the most similar image(s) to a given image.
- ☐ Given a triplet of images, the triplet loss uses two relations between them.
- ☐ There are loss functions that use all the relations in a mini-batch.
- ☐ Given an anchor  $i$ , hard-negative mining is the process of finding the negative examples that are most similar to  $i$ .

e) Which of the following is true for Message Passing Networks (check all that apply):

- ☐ They can be implemented as graph neural networks.
- ☐ They are invariant to node permutations.
- ☐ They can be trained end-to-end.
- ☐ They use node embeddings and might use edge embeddings.

f) Which of the following is true for Transformers (check all that apply):

☒ Transformers use an encoder-decoder architecture.

☒ Typically, a transformer layer has both multi-head attention and MLP sub-layers.

☒ Positional encoding is always learned in Transformers.

☒ The number of layers in the encoder does not need to be the same as the number of attention heads in the attention layers.

Sample Solution

## Problem 2 Generative models and trajectory prediction (48 credits)

0 ☐  
1 ☐  
2 ☐  
3 ☐

a) Write any loss function of Generative Adversarial Networks (GAN) (1p). What is generator G trying to maximize with respect to discriminator D (1p)? What is discriminator D trying to maximize with respect to generator G (1p)?

- Discriminator loss  $-0.5\mathbb{E}_x \log D(x) - 0.5\mathbb{E}_z \log(1 - D(G(z)))$ .
- Generator loss  $-0.5\mathbb{E}_z \log(D(G(z)))$ .
- Similar loss functions are also acceptable.
- The generator G is trying to maximize the (log) probability of the discriminator D being mistaken.
- The discriminator D is trying to maximize the probability of the generated output to be classified as fake.

0 ☐  
1 ☐  
2 ☐

b) Variational autoencoder contains two terms that need to be optimized. Briefly explain each of them and what is their task (1p for each).  
NB: It is enough to either write the objective of VAE, or describe it clearly.

In variational autoencoders, the loss function is composed of a reconstruction term (that makes the encoding-decoding scheme efficient) (1p) and a regularisation term, that makes the latent space regular (normal Gaussian). (1p) More mathematically, the reconstruction loss is the L2 loss of input and reconstructed output (alternatives are acceptable). The regularisation term is often the KL divergence of the latent distribution and the normal Gaussian (again alternatives are acceptable as long as it enforces a certain label distribution).

0 ☐  
1 ☐  
2 ☐

c) What is the main difference between BicycleGAN and Social BiGAT (1p). Explain it (1P).

Social BiGAT takes the idea of BicycleGAN and applies it to the task of trajectory prediction. It further adds a **social graph module**/ Graph Attention Network. The graph attention networks models social interactions.

d) Describe three key ideas behind PointNet, discussed in the lecture. (1p each)

<input type="checkbox"/>	0
<input type="checkbox"/>	1
<input type="checkbox"/>	2
<input type="checkbox"/>	3

- Per-point encoding (MLP)
- Symmetric, permutation invariant representation via max pooling
- T-net for invariance for transforms

e) What is the key feature of PointNets that ensures that learned representations are invariant to rigid transformation (1P)? Briefly describe what it does (1P).

<input type="checkbox"/>	0
<input type="checkbox"/>	1
<input type="checkbox"/>	2

- T-Net, which is a small PointNet network that estimates canonical pose and transforms the input point cloud.

f) Describe two key ideas behind PointNet extension, PointNet++ (1P each).

<input type="checkbox"/>	0
<input type="checkbox"/>	1
<input type="checkbox"/>	2

- It is applied recursively on a nested partitioning (downscaled) input point set.
- This way, we learn features with increasing contextual scales (similar to image counterparts, Multi-scale point-net).

0 ☐ g) Based on the lecture content, how would you design (with as little effort as possible, aiming at component reusability) as general as possible method for LiDAR 3D detection and tracking, that can be used in conjunction with LiDAR, stereo, or monocular data? Briefly explain.

1 ☐  
2 ☐  
3 ☐

- Based on stereo/monocular depth estimators, we can obtain a pseudo-lidar representation of a signal.
- From here on, as shown in the lecture, we can train a 3D object detector and tracker or use simple geometry/motion-only tracker, either is fine.
- (For correction: key message is to look for a unified representation).

0 ☐ h) Describe the task of panoptic segmentation (1P), and explain how it differs from traditional 3D amodal object detection (1P).

1 ☐  
2 ☐

- Panoptic segmentation: Semantic segmentation + instance segmentation.
- Difference: modal segmentation, per point/pixel classification (aka segmentation) instead of abstracting full object extend with 3D bounding boxes.

0 ☐ i) The original DeepLab uses Conditional Random Fields (CRFs). Describe the problem with this approach (1P) and mention a potential solution discussed in the lecture

1 ☐  
2 ☐

- Problem: not trained end-to-end, makes training both slow and arguably suboptimal.
- Solution: (a) Formulate CRF as an Recurrent Neural Network (CRF-RNN)
- (Also correct): CRFs look at all the pixels to improve masks (contours). Attention could be used instead. We could also let ASPP count.

j) FlowNet uses a network design called Siamese architecture to predict optical flow. Describe, what is optical flow (1p), what is the idea of a siamese architecture (1p), and the key layer that is used to combine information from different images and how it is different from a convolution (1p).

0  
1  
2  
3

- Input 2 images, output: displacement (perceived motion) of every pixel from first to second image (or vice versa)
- The same network (shared weights) is used to independently extract features for each of the input images.
- Correlation layer: The features of image 1 and image 2 are correlated, no weights are used in contrast to a convolution.

k) In proposal-based methods, it is common to use a technique called non-maximum suppression (nms). Given a set of bounding boxes B, implement the technique in pseudo code. You can consider the functions same and score implemented.

0  
1  
2  
3

```

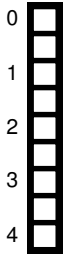
procedure same(box1, box2) return true iff box1 and box2 are the same object
procedure score(box1) return the score of box1.
procedure nms(B)
  B_nms <- empty
  //Implement and return B_nms
  
```

```

procedure nms(B)
  B_nms <- empty
  For b_i in B:
    discard <- False
    For b_j in B:
      If same(b_i, b_j) then
        If score(b_i) > score(b_j) then
          discard <- True
      If not discard then
        B_nms <- B_nms union b_i
  Return B_nms
  
```

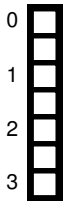
for b\_i in B:  
 discard <- false  
 for b\_j in B-n:  
 if same(b\_i, b\_j) >  $\lambda_{nms}$  then  
 if score(b\_i) > score(b\_j) . then  
 discard <- True  
 if not discard then  
 B\_nms <- B\_nms  $\cup$  b\_i  
 return B\_nms





l) You are given a set of object detections over frames over a video. Name two algorithms that can be used for data association with the given detections (0.5 each). What is the advantage of each of them (1p each)? Name two possible pairwise affinity features between detections that you could use as costs in your optimization problem (0.5p each).

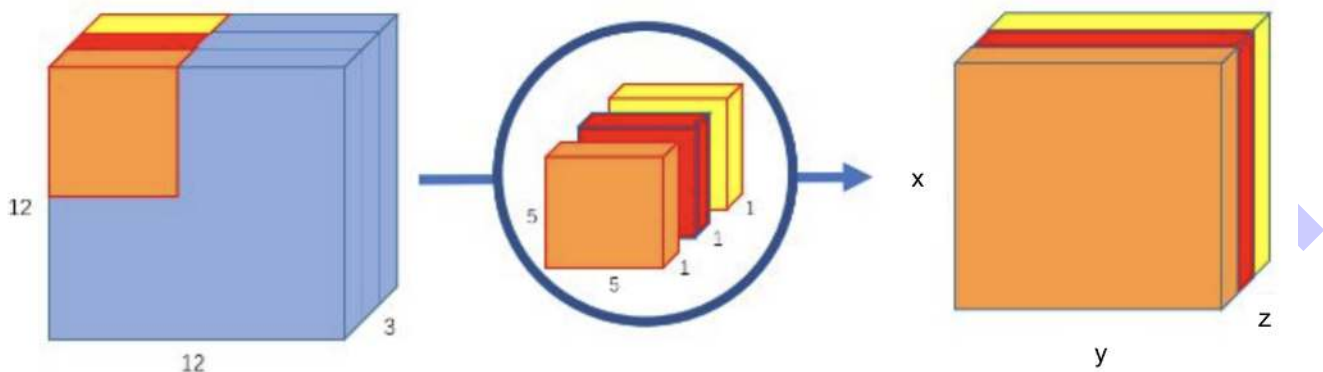
- Algorithms: Hungarian algorithm (Bipartite matching is also ok). Linear relaxation of Min-Cost Flow (just Min Cost Flow/Network flow /Commodity flow are also ok)
- Advantages: Bipartite matching runs online. Flow formulation can recover from missing detections in a given frame.
- Features: IoU, ReID similarity, Optical, Motion model + IoU, etc. Anything reasonable is fine.



m) Given a set of tracks from frame  $t-1$ , how does tracktor obtain new boxes for the given tracks at frame  $t$  (1p)? Make sure to name the Faster R-CNN head that it uses (1p). How does tracktor recover lost tracks (1p)?

- New boxes: place last box from track in frame  $t$ , then feed the corresponding RoIs features to **bounding box regressor** (1P) to obtain box coordinates at current frame.
- Recover from lost tracks: use reid similarity between lost tracks and unmatched detections (just saying using reid is also fine)

n) Given an input image of size 12 times 12 times 3 and applying on it three depthwise convolutional filters of size 5 times 5 times 1, write the dimension of the output feature vector. It is enough to write the values of x, y, and z in the figure. Assume that the convolutional stride is 1, and we apply no padding (1p each). Explain the difference between depthwise convolutional filters and vanilla (default) convolutional filters (2p).



- $x=8$  and  $y=8$  and  $z=3$
- Filters are applied on slices of the feature map. Normal convolutions in this case would have 3 dimensions (x, y, channel dimension) and produce a single output.
- Depthwise convolution kernels have 2 dimensions (x, y) and are applied on the corresponding channel.
- For the same output size, depthwise convolutions need much lower number of parameters compared to a normal convolution.
- (Remark, not required: Students might also point out that implementationwise they are equivalent to grouped convolutions with groups equal to channel dimension, here 3)

o) Given a sequence with length  $n$ , and a representation dimension of size  $d$ : what is the complexity of self-attention operator (1p), what is the complexity of RNN (1p)? Which one would be faster for machine translation, justify (1p)?

- Self-Attention  $\mathcal{O}(n^2d)$
- RNN  $\mathcal{O}(nd^2)$
- In most cases the sequence length  $n$  will be significantly lower than the representation dimension  $d$ . Thus, self-attention would be faster for machine translation.

0 ☐

1 ☐

2 ☐

p) There are several training tricks to make Transformers work faster. Describe two of them (1p each).

- Residual dropout.
- Label smoothing.
- Checkpoint averaging.
- Adam optimizer with proportional learning rate.

0 ☐

1 ☐

2 ☐

q) We have given the main formula in the self-attention.  $\text{softmax}(QK^T)V$ . However, there is a missing term in the equation. Complete the equation (1p). What is the purpose of the missing term (1p)?

- The correct equation is  $\text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$ . Thus division by  $\sqrt{d_k}$  is missing.
- Due to the large values in the key dimension, the dot product grows large in magnitude, pushing the softmax function into regions which lead to extremely small gradients.

0 ☐

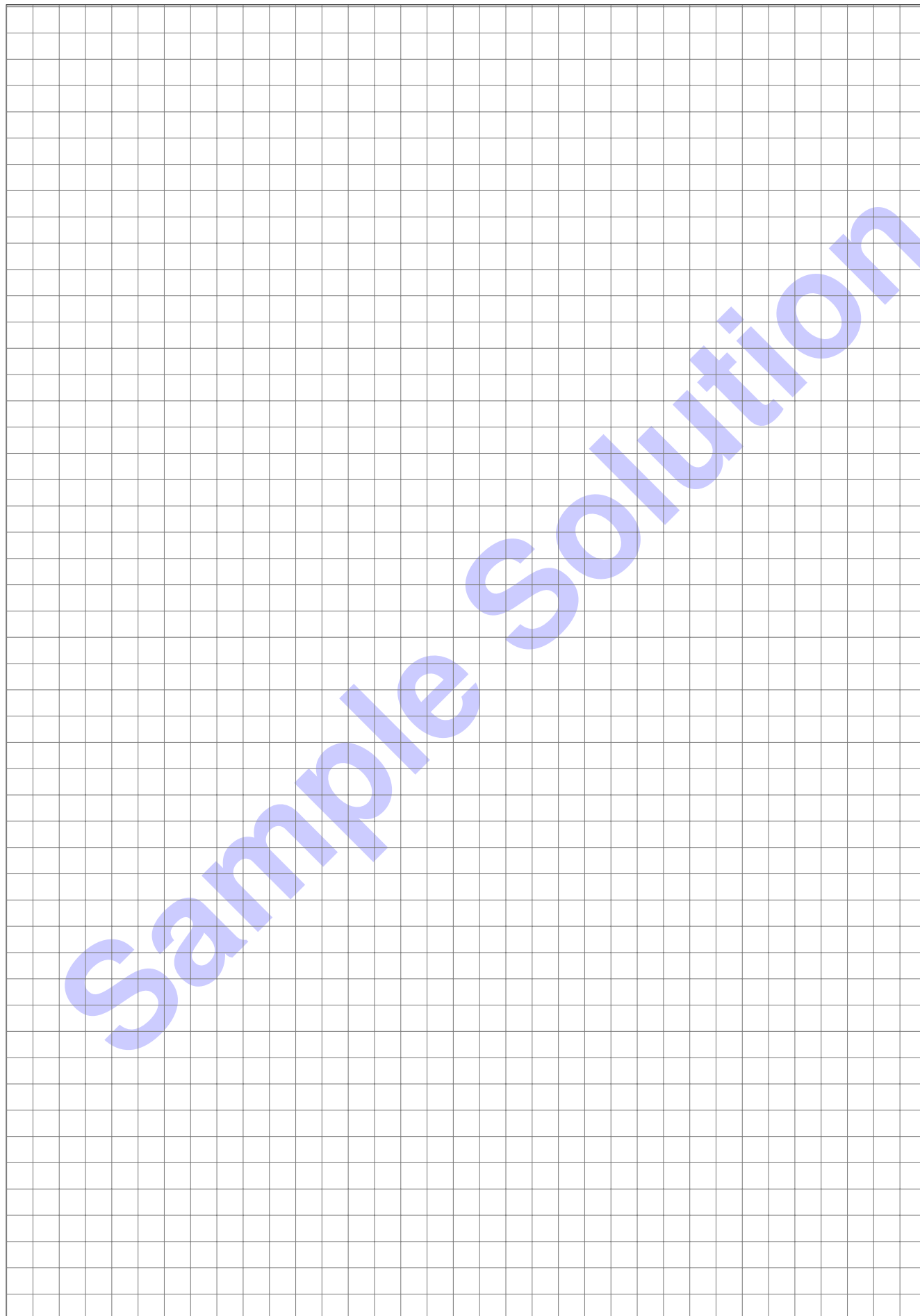
1 ☐

2 ☐

r) Describe in detail how mIoU is computed.

- Explains intersection (overlapping area of two segments/boxes of the same class).
- Explains union (combined area of two segments/boxes of the same class).
- IoU is computed per class -> the mean is used to get the average over the number of classes.

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

A large grid of graph paper, consisting of 30 columns and 30 rows of small squares, intended for writing solutions. A large, light blue diagonal watermark reading "Sample Solution" is overlaid across the grid.

Sample Solution