**Final project topics**

[**Object recognition and computer vision 2024**](https://imagine.enpc.fr/~varolg/teaching/recvis24/)

*Please read the full instructions (especially before sending any email)*

**Description:**

The final project amounts to 50% of the final grade. You will have the opportunity to choose your own research topic and to work on a method recently published at a top-quality computer vision conference ([ECCV](https://www.ecva.net/index.php#conferences), [ICCV](https://iccv2023.thecvf.com/), [CVPR](https://cvpr2023.thecvf.com/)) or journal ([IJCV](https://www.springer.com/journal/11263), [TPAMI](https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=34)). We also provide a list of interesting topics / papers below. If you would like to work on another topic (not from the list below), which you may have seen during the class or elsewhere, please consult the topic with the class instructor (Gül Varol <gul.varol@enpc.fr>) before submitting the project proposal. You may work on a similar topic as in another class project, in this case, you need to clarify which part of the project will be performed for this class. You may work alone or in a group of maximum 2 people. If working in a group, we expect a more substantial project, and an equal contribution from each student in the group.

For any chosen project, whenever possible, you should aim to **reproduce** published results using comparable experimental settings. If you use existing code and pretrained models, you can expect to obtain results similar to the published ones. In other situations, e.g., if you retrain a model using limited training data or train the model for a limited number of iterations, your results might be lower. This is OK, yet it’s still important to compare your results using the same test protocol as used by the others. Moreover, it is important to analyze results both **quantitatively** using metrics adopted for the task by others (accuracy, success rates,…) and **qualitatively** by visualizing test samples for which the method works well and for which it fails. Visualizing problematic test images/videos and corresponding results is a key to get ideas about further improvements.

Your task will be to:

(i) read and understand a research paper,

(ii) implement (a part of) the paper,

(iii) try a small **extension to the existing method**,

(iv) perform qualitative/quantitative experimental evaluation.

The final projects typically require multiple weeks of intense work, therefore, starting early is recommended.

**Evaluation and due dates:**

* **Project proposal (due on Nov 19).** You will submit a 1-page project proposal indicating (i) your chosen topic, (ii) the plan of work, i.e., what you are going to implement, what data you are going to use, what experiments you are going to do, (iii) if working in a group, who are the members of the group and how you plan to share the work. The project proposal will represent 10% of the final project grade.
* **Project presentation (Jan 6 and/or 7)**. You will present your work, with a quick introduction of the topic, followed by your contributions and experiment results. The project presentation will represent 20% of the final project grade.
* **Project report (due on Jan 13).** You will write a short report (3 double-column pages in CVPR format -- see below) summarizing your work. The report will represent 70% of the final project grade.

**Collaboration policy for final projects**

You can discuss the final projects with other students in the class. Discussions are encouraged and are an essential component of the academic environment. We encourage you to work **in a group of 2 people**. If working in a group, we expect a more substantial project, and an equal contribution from each student in the group. The final project report needs to explicitly specify the contribution of each student. Both students are expected to present the project at the oral presentation and contribute equally to writing the report. *The final projects will be checked to contain original material. Any uncredited reuse of material (text, code, results) will be considered as plagiarism and will result in zero points for the assignment / final project. If plagiarism is detected, the student will be reported to MVA. See below the policy on re-using other people's code.*

**Re-using other people’s code:**

You can (and in most cases should) re-use other people’s code. However, **you must clearly indicate in your report/presentation, what is your own code and what was provided by others (don’t forget to indicate the source)**. We expect projects balanced between implementation / experimental evaluation. For example, if you implement a difficult algorithm from scratch, only a few qualitative experimental results may suffice. On the other hand, if you completely use someone else’s implementation, we expect a strong quantitative experimental evaluation with analysis of the obtained results and comparison with baseline methods.

**Instructions for writing and submitting the project proposal:**

* You will submit a **1-page project proposal** indicating (i) your chosen topic, (ii) the plan of work, i.e., what you are going to implement, what data you are going to use, what experiments you are going to do, (iii) who are the members of the group and how you plan to share the work. The due date for the proposal is given at the beginning of this page. The project proposal should be a single 1-page pdf file. The proposal should go beyond the topic descriptions below and be more detailed.
* The proposal pdf should be named using the following format: FP**P**\_topicno\_lastname1[\_lastname2].pdf, where you replace "lastname\*" with last names of all members of your group in an alphabetical order, and replace “topicno” with the letter corresponding to your chosen topic. For a group consisting of 2 people: G. Varol, I. Laptev, choosing topic B, the file name should be **FPP\_B\_Laptev\_Varol.pdf**. For a single person project, with your own chosen topic, the file should be named **FPP\_X\_Varol.pdf**.
* **Upload your proposal to Google Classroom before the due date.**

**Instructions for writing and submitting the final project report:**

* You will hand-in a **3-page report** in the format of the submission to the [IEEE Computer Vision and Pattern Recognition conference (CVPR)](https://cvpr.thecvf.com/). Use the latex or word templates provided at the [CVPR Author Guidelines webpage](https://cvpr.thecvf.com/Conferences/2024/AuthorGuidelines). Note, that you are asked to produce **only a 3-page double-column report** (in contrast, a standard CVPR submission is up to 8 pages). Figures with qualitative results such as visualization of image classification or object detection results as well as references will **not** count to the 3-page limit, we encourage you to show visual results. For dynamic video results, you can also include links to videos from the report.
* At the top of the first page of your report should include (i) the title of your final project, along with the topic number, (ii) names of all members of your group, and (iii) the date.
* The report should be a single pdf file and should be named using the following format: FP**R**\_topicno\_lastname1[\_lastname2].pdf, similar to proposal naming convention described above.
* **Upload your report to Google Classroom before the due date.**

**Instructions on preparing the project presentation.**

* Each group will present their final project work.
* **Timing.** The exact timing and schedule of the presentations will be determined during the course.
* **Who should speak?** You can have one person presenting for the whole group, but it is preferable that all members of the group get to present a part of the project.
* **Content.**  You should introduce the topic, clearly state what the goal of the project is. Show the work you have done, clearly state your own contributions (i.e., what did you implement and experiment). Compare your results to results of previous methods obtained in comparable settings when possible. When reproducing previous methods, make sure to compare to original results. When describing results, please show both qualitative and quantitative results you have obtained and any interesting observations / findings you have made. Your audience are the class instructors, your project supervisor, and potentially the other students in the class. You want to show us that you have done interesting work. Remember, it is good to illustrate your findings with images/videos.
* **Reusing material / figures / slides from other people.** You can take figures from papers or other people’s slides to illustrate an algorithm or explain a method. However, always properly acknowledge the source if you do so.
* **Submitting slides.** Exact instructions for submitting your presentation will be given later.

**Suggested final project topics:**

[Topic A - Composed Image Retrieval 3](#_gzc89wbm5red)

[Topic B - 3D Human Motion Generation 4](#_qaa1famigte)

[Topic C - Generalizable Vision-Language Robotic Manipulation 6](#_5ccsum8pucwi)

[Topic D - TokenCompose: Text-to-Image Diffusion with Token-Level Supervision 7](#_dr77815sypax)

[Topic E - Test-Time Training with Masked Autoencoders 8](#_x0ru7nurvo8r)

[Topic X - Your own chosen topic 9](#_3mieced391pl)

## Topic A - Composed Image Retrieval



**Proposed by:** Lucas Ventura <[lucas.ventura@enpc.fr](mailto:lucas.ventura@enpc.fr)> / Gul Varol <[gul.varol@enpc.fr](mailto:gul.varol@enpc.fr)>

**Motivation:** Composed image retrieval is an emerging task in computer vision that involves retrieving target images by specifying both a query image and a text description of the desired modifications to that image.

**Description:** The goal of this project is to develop and evaluate models for the composed image retrieval task on the CIRR [2] dataset. CIRR contains triplets of (query image, modification text, target image).

**Dataset:** CIRR dataset - You will need to request access [here](https://docs.google.com/forms/d/e/1FAIpQLSdB_OhgmpQULV17kjQ4iitftILbOJjuGgJ2ECmg-HdmkjUSAg/viewform) (do this as soon as possible).

**Code:** <https://github.com/lucas-ventura/CoVR>

**Project Plan:**

1. Read [1].
2. Reproduce the baseline results on the CIRR dataset from [1] ([Table 3](https://arxiv.org/pdf/2308.14746.pdf), the row without WebVid-CoVR pretraining: 50.43 R@1) using the available code [here](https://github.com/lucas-ventura/CoVR).
3. Train with 1 GPU and lower the batch size, this will decrease performance, but you will iterate much faster. This new result will be your baseline from now on.
4. Try to improve upon these results with one (or more) direction from below:
   1. The method in [1] uses the combined multi-modal embedding f(q, t) (i.e., output of shift encoder in the code) to retrieve the target image. First, replace the query with the average of three embeddings: f(q, t), image embedding q, and text embedding t. Then, try adding a small neural network (e.g., a few-layer MLP) to output the weighting between the three embeddings: w1\*multimodal\_emb + w2\*img\_emb + w3\*txt\_emb. You can experiment with different variants. The motivation is to check if we lose text or image information when computing the multi-modal embedding. Also, there are a few examples where only the text or image is important, so weighting the two information may be interesting.
   2. With the [WebVid-CoVR dataset](https://imagine.enpc.fr/~ventural/covr/), create a subset of triplets where query image and target video have come from the same author or same category (check [shinonomelab/cleanvid-15m\_map](https://huggingface.co/datasets/shinonomelab/cleanvid-15m_map) for this metadata). Train with this subset to check if this helps removing some noise, assuming that same video owners have visually similar recordings.
   3. Try training with hard negatives. Modify the dataloader to sample negatives from the same group (in CIRR, these are called “members” in the split json files). Ideally, you should sample all members of the same group in the same batch. Also experiment with filtering them so there all the triplets come from different members.
   4. Train the model on the full CC-CoIR dataset, then explore techniques to reduce the dataset size while maintaining performance. You can read the [SemDeDup](https://arxiv.org/pdf/2303.09540) paper for inspiration. This might be more compute-expensive.
   5. Introduce a consistency loss between different modalities (image, text, multimodal). This could enforce that the learned multimodal representation agrees with its constituent parts (e.g., the text should still describe the image even after they are combined in a multimodal embedding).
   6. Experiment with different pooling techniques to aggregate the 32 learnable queries embeddings, such as max-pooling or attention-based pooling, rather than simple averaging.
   7. Implement techniques to visualize and interpret what the model is attending to when making predictions. This could provide insights for further improvements.
   8. Come up with your own improvement idea.
5. Evaluate results on CIRR to see whether or not there is an improvement.
6. Visualize and analyze failure modes.

**References:**

[1] [CoVR: Learning composed video retrieval from web video captions](https://arxiv.org/abs/2308.14746). Lucas Ventura, Antoine Yang, Cordelia Schmid, and Gul Varol. AAAI, 2024.

[2] [Image retrieval on real-life images with pre-trained vision-and-language models](https://arxiv.org/abs/2108.04024). Zheyuan Liu, Cristian Rodriguez-Opazo, Damien Teney, and Stephen Gould. ICCV 2021.

## Topic B - 3D Human Motion Generation

## 

**Proposed by:** Léore Bensabath <[leore.bensabath@enpc.fr](mailto:leore.bensabath@enpc.fr)> / Gul Varol <[gul.varol@enpc.fr](mailto:gul.varol@enpc.fr)>

**Motivation:** 3D human motion generation is a tedious task requiring expertise when implemented deterministically, for instance by an animator. Being able to generate realistic and accurate motion from natural language could make a substantial difference in fields such as films and video games, and even put motion generation capability within the reach of beginner animators.

**Description:** The goal of this project is to evaluate and improve motion generation models using data augmentation. Text augmentation has proven useful for the text-to-motion retrieval task [1]. You will assess the effect of this same text data augmentation for motion synthesis using a diffusion model, then will be free to explore other ideas to improve the model.

**Dataset:** Texts:[KIT-ML dataset](https://motion-annotation.humanoids.kit.edu/dataset/) - [HumanML3D dataset](https://github.com/EricGuo5513/HumanML3D) - [BABEL dataset](https://babel.is.tue.mpg.de/); Motions: [AMASS](https://amass.is.tue.mpg.de/)

Unifying text datasets: [AMASS-Annotation-Unifier](https://github.com/Mathux/AMASS-Annotation-Unifier)

**Code:**

* TMR++[1] <https://github.com/leorebensabath/TMRPlusPlus> (data)
* STMC [3] <https://github.com/nv-tlabs/stmc>(MDM-SMPL model)

**Project Plan:**

1. Read [1], [2] and [3]
2. Download the datasets and set up the code fromSTMC [3]. You will mainly use the MDM-SMPL component, that extends MDM [2] with the SMPL body model. This project does not need to involve “multi-track timeline” component, but primarily focuses on evaluating single-text as input.
3. Download the datasets and set up the code fromTMR++ [1]. You don't need to set up motion and checkpoint data from here, since you’ll use the ones from the STMC repo. Only download the annotations from TMR++.
4. Evaluate MDM-SMPL pretrained models (trained on HumanML3D) fromthe STMC repo:
   1. test on HumanML3D
   2. test on KIT-ML
   3. Optional: test on BABEL
5. Use text-augmented dataset fromTMR++ (transfer the annotation folders in STMC repo) to train MDM-SMPL. Run the following experiments:
   1. Train on KIT-ML → test on HumanML3D; test on KIT-ML;
   2. Train on HumanML3D + KIT-ML → test on HumanML3D; test on KIT-ML;
   3. Train on HumanML3D + KIT-ML with text augmentation → test on HumanML3D; test on KIT-ML;
   4. Optional: include BABEL
6. [optional] Try to improve the generalization performance of MDM-SMPL via motion data augmentation, like dropping/repeating some frames, varying the fps, adding noise, encoding-decoding with added noise in the bottleneck etc.
7. [optional] Generate synthetic data with STMC of 2-3 compositions, train on single motions from this training data, test with STMC evaluation data.
8. [optional] Come up with your own improvement idea.
9. Showcase with visualizations which type of texts/motions work better.

**References:**

[1] [A Cross-Dataset Study for Text-based 3D Human Motion Retrieval](https://arxiv.org/abs/2405.16909). Léore Bensabath, Mathis Petrovich, and Gül Varol. CVPRW 2024. [[code](https://github.com/leorebensabath/TMRPlusPlus)]

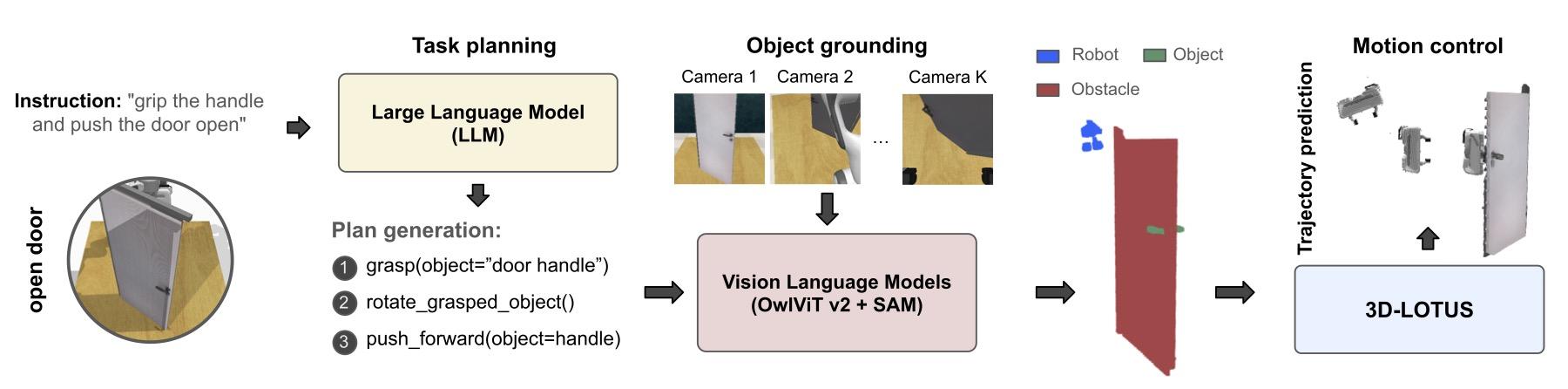
[2] [Human Motion Diffusion Model](https://arxiv.org/abs/2209.14916). Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or and Amit H. Bermano., ICLR 2023. [[code](https://github.com/GuyTevet/motion-diffusion-model)]

[3] [Multi-Track Timeline Control for Text-Driven 3D Human Motion Generation](https://arxiv.org/abs/2401.08559). Mathis Petrovich, Or Litany, Umar Iqbal, Michael J. Black, Gül Varol, Xue Bin Peng, Davis Rempe. CVPRW 2024. [[code](https://github.com/nv-tlabs/stmc)]

## 

## 

## Topic C - Generalizable Vision-Language Robotic Manipulation

****

**Proposed by:** Ricardo Garcia <[ricardo-jose.garcia-pinel@inria.fr](mailto:ricardo-jose.garcia-pinel@inria.fr)>

**Motivation:** A long-term goal in robotics is to develop robots capable of following language instructions to perform diverse manipulation tasks, though generalizing language-conditioned robotic policies to novel tasks remains a major challenge. A recent work introduces GemBench [1], a novel challenging benchmark based on RLBench [6] to assess generalization capabilities of vision-language robotic manipulation policies. GemBench incorporates seven general action primitives and four levels of generalization, spanning novel placements, rigid and articulated objects, and complex long-horizon tasks. State-of-the-art methods [2], [3], [4], [5] attain good performance on the same multi-tasks where they are trained on showing generalization to novel placements but failing to generalize on more difficult levels of generalization. A new approach, 3D-LOTUS++ [1] integrates a point cloud based method 3D-LOTUS’s motion planning capabilities with the task planning capabilities of LLMs and the object grounding accuracy of VLMs to attain generalization. 3D-LOTUS++ achieves state-of-the-art performance on novel tasks of GemBench, setting a new standard for generalization in robotic manipulation.

**Description:** The goal of this project is to explore the 3D-LOTUS++ method and extend it to improve its performance on different levels of the robotic manipulation task generalization from the GEMBench benchmark [1].

**Dataset:** a set of GEMBench [1] expert demonstrations for the training 31 variations across 16 tasks (100 expert trajectories per task variation) and the initial state information for validation and test task variations on 5 different seeds (20 configurations per task variation). The dataset can be found [here](https://www.dropbox.com/scl/fo/y0jj42hmrhedofd7dmb53/APlY-eJRqv375beJTIOszFc?rlkey=2txputjiysyg255oewin2m4t2&st=vfoctgi3&dl=0).

**Simulator:** GEMBench framework [1] from [this GitHub repository](https://github.com/rjgpinel/RLBench).

**Project Plan:**

1. Carefully read [1]. Understand GEMBench benchmark, 3D-LOTUS and 3D-LOTUS++ methods.
2. Install GEMBench [RLBench simulator version from [1]](https://github.com/rjgpinel/RLBenchMohitShridhar/RLBench/tree/peract) and 3D-LOTUS++ code.
3. Get familiar with 3D-LOTUS++ and GEMBench code reproducing 3D-LOTUS++ results on a subset of tasks from Tables X, XI, XII, XIII [1] by using the 3D-LOTUS++ official codebase and pretrained model from [1]. Analyze the existing failure cases and method limitations. Note: Given the resources limitations, only run test on one unique seed and a selected subset of tasks.
4. Train 3D-LOTUS++ model on the provided expert demonstrations and report its performance. Analyze the existing failure cases and method limitations.
5. Extend 3D-LOTUS++ by one (or more) of the following options:
   1. Investigate the task planning module. Is there any way to extend this module to also process the visual observations at each sub-plan execution to adapt to errors and allow replanning?
   2. Investigate the object grounding module. Check alternative VLMs models or find any directions to improve this module.
   3. Finetune the task planning module on new data or object grounding module with the simulator data.
   4. Come up with your own extension ideas to improve the performance.

**References:**

[1] [Towards Generalizable Vision-Language Robotic Manipulation: A Benchmark and LLM-guided 3D Policy](https://arxiv.org/abs/2410.01345). Ricardo Garcia-Pinel, Shizhe Chen, and Cordelia Schmid. CoRL 2023. [[code](https://github.com/cshizhe/robot-3dlotus)]

[2] [Instruction-driven history-aware policies for robotic manipulations](https://arxiv.org/abs/2209.04899). PL. Guhur et al.CoRL 2022. [[code](https://github.com/vlc-robot/hiveformer)]

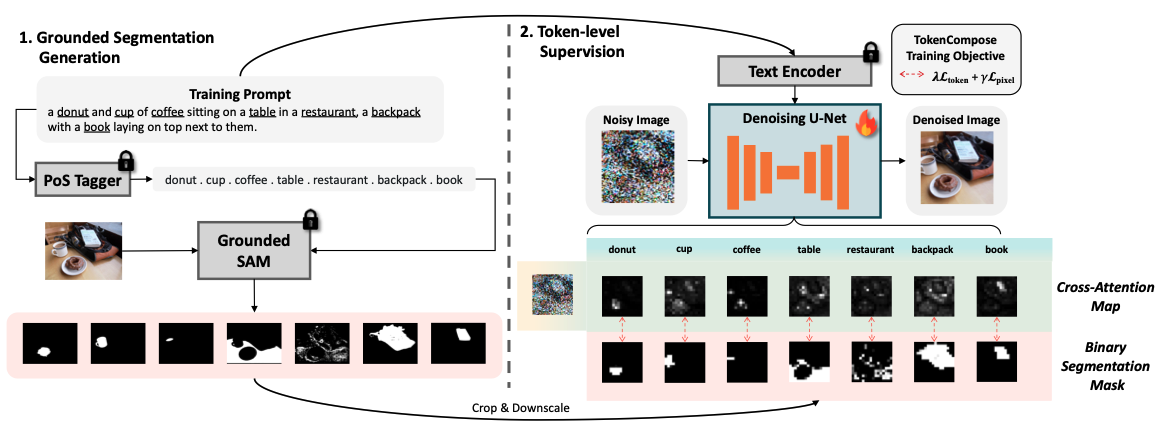
[3] [PolarNet: 3D Point Clouds for Language-Guided Robotic Manipulation](https://openreview.net/forum?id=efaE7iJ2GJv). Chen S et al. CoRL, 2023. [[code](https://github.com/vlc-robot/polarnet/)]

[4] [3D diffuser actor: Policy diffusion with 3D scene representations](https://arxiv.org/abs/2402.10885). Tsung-Wei Ke et al. arXiv 2024. [[code](https://github.com/nickgkan/3d_diffuser_actor.git)]

[5] [RVT-2: Learning precise manipulation from few demonstrations](https://arxiv.org/abs/2406.08545). Ankit Goyal et al. RSS 2024. [[code](https://github.com/NVlabs/RVT)]

[6] [RLBench: The Robot Learning Benchmark & Learning Environment](https://arxiv.org/abs/1909.12271). S. James et al. ICRA 2020. [[code](https://github.com/stepjam/RLBench)]

## Topic D - TokenCompose: Text-to-Image Diffusion with Token-Level Supervision



**Proposed by**: Zeeshan Khan <[zeeshan.khan@inria.fr](mailto:zeeshan.khan@inria.fr)>

**Motivation**: Text to image (T2I) diffusion models have achieved tremendous success in terms of high quality generation with good text following abilities. However state-of-the-art T2I models like Stable Diffusion [1] (versions 1, 2, and, SDXL) struggle to follow the text prompt for composing multiple object categories. The standard denoising process in the latent diffusion model takes text prompts as conditions only, i.e., lacking an explicit constraint for the consistency between the text prompts and the image contents. This leads to unsatisfactory results for composing multiple object categories. This project looks into improving the multi-object compositional generation abilities of T2I models.

**Description**: A recent approach - TokenCompose [2] aims to improve multi-object instance composition by introducing object word level consistency terms between the image content and object segmentation maps. Text is conditioned in SD 1.4 via cross attention between image latent and token embeddings of the text prompt. TokenCompose extracts token-level attention maps which describe the semantic closeness of the latent pixels and the word token. TokenCompose trains the cross attention maps using the ground-truth segmentation maps that introduce extra constraints by focusing on each object. TokenCompose uses 2 losses, pixel loss and token loss. The goal is to explore and analyze the problem of multi-object image generation and how to mitigate it via TokenCompose [2].

**Project Plan:**

1. Read [1, 2] get familiarized with Latent Diffusion Models (SD 1.4) [1] and TokenCompose [2].
2. Set up the codebase of Tokencompose [2] and download the dataset that contains around 4k images from MS COCO along with their captions, and their corresponding segmentation maps from [here](https://github.com/mlpc-ucsd/TokenCompose?tab=readme-ov-file#%EF%B8%8F-dataset-setup). All the instructions are clear and easy to follow.
3. [Inference] Evaluate SD 1.4 on the [MultiGen Benchmark](https://github.com/mlpc-ucsd/TokenCompose/blob/main/multigen/readme.md) report quantitative and qualitative results, following the [eval scripts](https://github.com/mlpc-ucsd/TokenCompose/tree/main/multigen).
4. [Inference] Evaluate the pretrained checkpoint on the [MultiGen Benchmark](https://github.com/mlpc-ucsd/TokenCompose/blob/main/multigen/readme.md) report quantitative and qualitative results, following the [eval scripts](https://github.com/mlpc-ucsd/TokenCompose/tree/main/multigen).
5. [Training] Train SD1.4 following the [train script](https://github.com/mlpc-ucsd/TokenCompose/blob/main/train/src/train_token_compose.py).
6. [Inference] Conduct an evaluation of the trained model on the MultiGen Benchmark.
7. Investigate the 2 new loss functions - pixel loss and token loss, and see how it affects the performance when one or both of them are removed.
8. Visualize the cross-attention maps of objects in a text prompt before and after training. See how the attention maps evolve after training with 1) pixel loss 2) token loss and 3) both pixel and token losses.
9. Come up with your own extension idea and check if the performance is improved.
   1. For example, you can add another loss for better alignment between tokens and their regions. You can apply an image-text contrastive loss between the CLIP token embedding and its corresponding region image feature.
      1. Let's say there are 5 word tokens and 5 segmentation masks per image. For each token extract the CLIP token embedding.
      2. Then extract the ROI image embeddings using a ROI pooling layer from the 16x16 feature map. This is the positive pair.
      3. Do the same with the next 4 tokens and their regions.
      4. Create negative pairs and apply per image contrastive loss.
   2. You can try replacing the SD1.4 backbone with the relatively small CAD model from [3].

**References:**

[1] [High-Resolution Image Synthesis with Latent Diffusion Models](https://arxiv.org/abs/2112.10752). Robin Rombach et al. CVPR 2022.

[2] [TokenCompose: Text to Image Diffusion with Token Level Supervision](https://arxiv.org/abs/2312.03626), Zirui Wang et al. CVPR 2024.

[3] [Don’t drop your samples! Coherence-aware training benefits conditional diffusion](https://arxiv.org/abs/2405.20324), Nicolas Dufour, Victor Besnier, Vicky Kalogeiton, David Picard. CVPR 2024. [[code](https://github.com/nicolas-dufour/CAD)]

## Topic E - Test-Time Training with Masked Autoencoders



**Proposed by:** Shizhe Chen <[shizhe.chen@inria.fr](mailto:shizhe.chen@inria.fr)>

**Motivation**: Masked autoencoders (MAE) have been shown very successful for unsupervised pretraining of large language models [3], and more recently for training computer vision models [2]. The use of unlabeled data in MAE enables its application to very large datasets during training. Instead of using MAE during training, [1] proposes to apply MAE to test images with the aim of adapting existing network to the target image distribution. The goal of this project is to explore and analyze this Test-Time Training (TTT) technique.

**Code:** [TTT-MAE](https://github.com/yossigandelsman/test_time_training_mae)

**Project plan:**

1. Read [1] and get familiar with the TTT-MAE approach.
2. Reproduce ImageNet-C results [1]. You can sample 1000 images out of the 10,000 per category to save computing.
3. Investigate images from the Imagenet validation set, in which the algorithm does not work, and characterize failure modes.
4. Investigate how the TTT-MAE number of steps affects the results.
5. Implement an online version of TTT-MAE, where at every new test image, the model is not reset to the original weights, but continues from the last checkpoint.
6. Investigate the influence from the order of images for the online version, e.g., continuously evaluate the images of the same category/corruption or evaluate images from different categories/corruptions in a random order.

**References**

[1] [Test-Time Training with Masked Autoencoders](https://arxiv.org/abs/2209.07522). Gandelsman et al., NeurIPS 2022. [[code](https://github.com/yossigandelsman/test_time_training_mae), [project page](https://yossigandelsman.github.io/ttt_mae/index.html)]

[2] [Masked autoencoders are scalable vision learners](https://openaccess.thecvf.com/content/CVPR2022/papers/He_Masked_Autoencoders_Are_Scalable_Vision_Learners_CVPR_2022_paper.pdf). He et al., CVPR 2022.

[3] [BERT: Pre-training of deep bidirectional transformers for language understanding](https://arxiv.org/abs/1810.04805). Devlin et al., NAACL 2019.

## Topic X - Your own chosen topic

You can also choose your own topic, e.g., based on a paper which has been discussed in the class, a topic proposed in previous years, etc. The supervision for custom topics may not be as guaranteed as above topics. Please validate the topic with the course instructor (Gül Varol <gul.varol@enpc.fr>) before submitting the project proposal.