The Chinese University of Hong Kong, Shenzhen
School of Data Science

MDS5122 / AIR5011
Final Project

Due by: 23:59, Dec 5th, 2025

Instructions:

1. You must submit your files on Blackboard. Please upload a PDF file along with your code (excluding large files such as the dataset).
2. Your submission must be clearly answered and well-presented to receive full credit. Please detail how you leverage your acquired knowledge to solve the problem and present your proposed solution step by step. Ensure that your solutions are legible and written in English.
3. The programming language is Python, and your code should include necessary comments so that others can easily follow its logic. Copying answers violates academic integrity.
4. Your report should reflect your individuality and originality. If your answers are inspired by other sources (*e.g.*, the Internet or AI), please maintain academic integrity by including a reference or acknowledgment section in your submitted paper and indicate how these sources inspired you.
5. Late submissions or instances of plagiarism will not be graded.

A. AI Model for Human-Object Interaction Frame Prediction (100 points)

Understanding human actions from short videos is an important frontier in multimodal deep learning. In this project, you will build a text-conditioned video prediction model: given several observed frames of a human–object interaction and a natural-language description of the ongoing action, your model must predict the next (or future) frame of that sequence.This task mimics the reasoning challenge of anticipating motion and visual change from linguistic and visual context—an ability relevant to robotics, AR/VR, and embodied AI.

Specifically, you will train an image-generation or video-prediction model to predict a future frame of a

human-object interaction. Given an observed frame from the Something-Something V2 dataset (*e.g.*, a person moving an object - the **move_object** task) and its original textual description (*e.g.*, "Moving something up"), your model should generate what the scene will look like 21 frames (frames ~~per second~~) later. The image size must be at least 96×96.

Your report MUST be written using the CVPR 2025 Author Kit in a camera-ready format, with your names, student IDs, and affiliations clearly displayed, and must adhere strictly to CVPR's requirements as if you were submitting to this conference, *e.g.*, a maximum of eight pages, although you may not actually submit it. You must also release and upload your code as a supplement to the Blackboard system, along with a Markdown document that clearly instructs others on how to set up the environment and reproduce your results. If you are confident in your project, you may optionally submit it to any other conference you like. Apart from the requirements stated above, there are no concrete restrictions on how your method should be implemented. If you cannot come up with your own ideas, you can follow the suggestions below to complete this final project.

Suggestions (Not Mandatory):

1. Dataset: Something-Something V2,Content: ~220K short human–object interaction videos (each video is about 2–6 s, ≈ 48 frames, 12 fps). Details in HuggingFace.
2. You can leverage a pretrained InstructPix2Pix model/Stable Diffusion plus ControlNet model as a base and fine-tune it for this task. Fine-tune the model to accept the current frame as well as a textual action instruction as inputs and predict a future frame.
3. For simplicity, this project focuses on three representative human-object interaction tasks: **move_object**, **drop_object**, **and cover_object**. Each task is treated as a sub-dataset, consisting of all videos whose textual labels match the selected categories (or their close variants). This partition allows students to perform per-task training, evaluation, and comparison—for instance, assessing how well a model predicts future frames in motion-dominant versus contact-dominant interactions.
4. Students are encouraged to extend the dataset to more action groups by selecting additional categories from Something-Something V2. This allows exploration of richer motion types and semantic diversity. (*e.g.*, open_object, throw_and_catch, pull_object).

Task Name	Example Category	Description
move_object	Moving something from left to right / Pushing something from right to left	The object undergoes a horizontal translation; used to model directional motion.
drop_object	Dropping something onto something / Letting something fall down	The object transitions from support to free-fall under gravity; used to model dynamic vertical motion.
cover_object	Covering something with something / Putting something on top of something	The agent manipulates an object to partially or fully cover another; used to model static contact completion.

5. The image size must be at least 96×96: A practical and efficient choice to guarantee successful project completion. Also you can challenge yourself with higher resolution (*e.g.*, 128×128): An excellent way to explore advanced optimization and elevate the quality of your work.
6. Use 20 observed frames as input and predict the 21th (or N+5) frame as output.
7. For each task, please generate at least 100 observations; the model can be fine-tuned using these 300 observations.
8. Evaluate using the SSIM (Structural Similarity Index) and PSNR (Peak Signal-to-Noise Ratio)

metrics.

9. Additionally, you may find GR-MG useful.
10. If you don't know what a typical CVPR paper looks like, you just need to refer to and emulate their writing style and the way they present their methods. For example, the ResNet paper is an excellent reference.

Scoring Criteria:

Your report/paper must contain the following sections:

1. Introduction **(5 points)**
2. Related Work **(2 points)**
3. Method **(25 points)**
4. Experiment **(50 points)**
5. Conclusion and Future Work **(3 points)**

Clarity of code and paper **(10 points)**

Summary of workload Distribution **(5 points)**

We will review and score each part of your paper as if you had submitted it to the CVPR conference. Using your own ideas, conducting ablation studies, performing comparative experiments, and so on means that you have a good chance of earning higher credits.

Some Tips on Experiments and Computational Resources:

0. **Adjusting Hyperparameters:** Consider adjusting hyperparameters, *e.g.*, model size, batch size, to ensure your experiments can be conducted within the anticipated time budget. You may include various techniques to reduce memory usage, such as half-precision training and quantized training. It is advisable to train using a GPU, with the experiment typically requiring a minimum of 12GB of GPU memory.
1. **NVIDIA GPUs:** If you have an NVIDIA GPU on your local machine, I recommend installing the CUDA version of PyTorch. This will allow the training process to utilize the GPU for acceleration.
2. **Online Free GPUs:** If you only have access to a CPU, you can explore online computational resources. *Google Colab* is an excellent platform for running deep learning experiments online. It is completely free to use, just sign in with a Google account. Similarly, you might consider *Kaggle* and *Tianchi*, both of which offer free GPU resources.
3. **University Resources:**Additionally, our school's high-performance computing platform is available, although registration is required and it is not free.
4. **Third-Party GPU Providers:** There are also several third-party GPU cloud providers, such as GpuShare and AutoDL, which offer affordable options, though they are not free.

If you have any issues, please don't hesitate to contact our TAs.