Final Project: Research Exploration in Computer Vision

Course: CS 5404 – Introduction to Computer Vision

Instructor: Dr. Ce Zhou

Proposal Due: November 7, 2025 (11:59 PM)

Final Report & Code Due: December 2, 2025 (11:59 PM)

Group Work: Up to 3 students per team (highly encouraged)

1. Objective

The final project aims to provide hands-on experience in research-oriented problem solving within the computer vision domain. You will explore cutting-edge topics by studying A* conference papers, reproducing or extending their work, and proposing your own idea built on top of existing research. This project encourages creativity, teamwork, and the ability to conduct reproducible and impactful vision research.

2. Project Overview

You will:

1. Select and read one or more research papers from top-tier A* computer vision conferences:
   - CVPR (Conference on Computer Vision and Pattern Recognition)
   - ICCV (International Conference on Computer Vision)
   - ECCV (European Conference on Computer Vision)
   - NeurIPS (Conference on Neural Information Processing Systems)
   - ICLR (International Conference on Learning Representations)
2. Run the official open-source implementation of the selected work (if available).
3. Propose a new idea that extends, improves, or applies the paper's method in a novel way.
4. Implement and evaluate your idea using code and datasets of your choice.
5. Write a research-style proposal and final report following the CVPR LaTeX format, describing your motivation, methodology, experiments, and findings.

3. Deliverables

A. Project Proposal (Due: November 7, 2025)

Submit a 2–3 page proposal in CVPR LaTeX format, including:

- Title and group member names (maximum 3 students)
- Introduction and motivation
- Summary of the base paper you are extending
- Problem statement and hypothesis
- Your proposed idea and preliminary plan
- Expected contributions and timeline

Tip: The proposal should clearly describe why your idea is interesting and how it differs from the original work.

B. Final Report and Code (Due: December 2, 2025)
Submit a 6–9 page report in CVPR LaTeX format along with your project code (**code should not be shown in the paper, submit separately**). The report must include:
- Abstract and introduction
- Related work (summary of the base paper and relevant literature)
- Methodology (your proposed improvement or extension)
- Experimental setup, dataset, and results
- Discussion and analysis
- Conclusion and future work

Code Submission:
- Include a link to your GitHub repository or zip file containing all source code.
- The code should be runnable with clear documentation and instructions.

Format requirement:
You must use the official CVPR LaTeX template available on Canvas. Overleaf usage is encouraged for collaborative editing.

C. Presentation
Each team will give a short presentation (10–12 minutes) summarizing their project, including background, methodology, results, and future directions. The presentation is mandatory and will be graded based on clarity, depth, and visual quality.

4. Collaboration
Working in teams is highly encouraged, but individual projects are also accepted. **Each group should submit one proposal, one final report, and one code package. (Only one person should submit these on Canvas.)** All members will receive the same grade unless contributions are clearly imbalanced (must be stated in the report).

5. Expectations and Research Outcome
By the end of the semester, we expect your project to reach the level of a conference-style paper draft. Outstanding projects will be invited to continue research with the instructor after the semester, with the goal of submitting a paper to a top-tier venue (e.g., CVPR, ICCV, ECCV, or NeurIPS).

6. Grading Breakdown (Total 400 points)

Component	Description	Points
Proposal Submission	Clarity of motivation, research plan, and feasibility	100
Final Report & Code	Technical quality, implementation correctness,	200

	and writing	
Presentation	Clarity, technical depth, and communication skills	100

7. Tips

• **Choose a paper that provides open-source code for reproducibility**.
• Focus on small but meaningful extensions, such as improving model robustness, applying the method to a new dataset, or simplifying computation.
• Clearly describe both what you tried and what you learned, even if the results differ from your expectations.
• **Manage your time early — start reading and coding as soon as possible.**