ENGR 430 Computer Vision

Students assemble their own flying quadcopter with a Raspberry Pi controller. The drone employs a camera and computer vision to determine its position and navigate autonomously. Student learn the theory of computer vision and experiment with software that extracts features from camera images. (3 credits). Special Offering Fall 2019

**Prerequisites**: None

**Textbook**: None

**Instructor:** Dr. Matthew R. Stein, SE106 254-3489. mstein@rwu.edu

**LEARNING OUTCOMES:** By the end of this course, students should be able to:

1. Employ fundamental terminology related to computer vision
2. Access and manipulate binary image data using the Python programming language
3. Write and comprehend software to extract features from image data
4. Comprehend the nature of visual features and the limitations of feature detection
5. Write and comprehend software to categorize and catalog feature data
6. Write and comprehend software to match features in two images
7. Write and comprehend software to determine camera location and orientation from matched feature sets
8. Employ fundamental terminology related to quadcopter components
9. Employ sufficient mechanical and electrical technology skills to construct a working quadcopter
10. Employ the Raspberry Pi as a general purpose computing platform
11. Use basic commands to navigate the Linux operating system

**TOPICS COVERED:**

1. Introduction to computer vision, camera models, digital representation of light
2. Homogeneous transformations and the representation of position and orientation
3. Image pyramids and filters for feature detection
4. Description of features and feature matching
5. The Linux operating system and the Raspberry Pi computing Platform
6. Fundamentals of quadcopter flight, power and control of quadcopters
7. Electrical fabrication techniques including soldering, wiring and mechanical contruction

**Class Schedule:** Class meets two times weekly for 80 minutes.

**Relationship of course to program LEARNING OUTCOMES:**

This class specifically responds to some degree to all program outcomes.

6. an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions

7. an ability to acquire and apply new knowledge as needed, using appropriate learning strategies.

|  |  |  |
| --- | --- | --- |
| GRADING: | Percent of total | Grade |
| Project 200 points | 93.33% | A |
| Homework 90 points | 90% | A- |
| In-class assessments 30 points | 86.67% | B+ |
|  | 83.33% | B |
|  | 80% | B- |
|  | 76.67% | C+ |
| Total 320 points | 73.33% | C |
| Table represents the minimum grade that will be | 70% | C- |
| assigned. Instructor reserves the right to “curve” | <70% | F |
| grades upwards, when appropriate, but will not | The grade of ‘D’ is not available in core engineering courses | |
| assign grades lower than those listed in this table. |

# Homework Format:

Most homework assignments will be submitted electronically via Bridges.

**Disability Statement:**

Students with disabilities who wish to receive academic accommodations for this course must register with Disability Support Services (DSS) in order to begin the accommodation process.   The DSS office will provide registered students with the specific information they will need to share with each instructor.  DSS is located on the second floor of the library.

# Assistance/Collaboration Policy:

Students may collaborate on homework assignments. Students must construct and demonstrate quadcopter performance individually.

# School of Engineering Attendance and Lateness Policy:

Regular attendance in classes is expected of all students. Faculty members are in no position to assess the validity of excuses and the Student Health Service will not provide notes or vouch for student health. Three absences, or one week of class, are allowed to all students for *any and all reasons*. No explanation is necessary for these three absences. Absences above three affect the final course grade according to the following schedule.

Unexcused absences:

|  |  |
| --- | --- |
| Up to 2 absences | No effect on final grade |
| 3 Absences | Final grade lowered by ½ letter grade. |
| 4 Absences | Final grade lowered by 1 letter grade. |
| 5 or more absences | Final grade of 0 assigned (Official withdrawal recommended). |

Attendance is typically monitored using "sign-in sheets" for each class meeting. It is the responsibility of the student to sign-in during each class attended. A student is considered absent of his or her signature does not appear on the sign-in sheet. Forgery of signatures on sign-in sheets is a violation of university Academic Integrity Policies and subject to disciplinary action.

Tardiness: Each three instances of excessive tardiness (e.g. > 10 minutes) will count as one absence.

Late assignments: Assignments are submitted via Bridges and date-stamped on submission. Students may submit one homework assignment late without penalty. All late assignments must be turned in *within one week of the due date and by the last day of class*.

**Detailed Class Schedule:**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Date** | Due | Content/Activity |
|  | Aug 29 (Th) |  | Course structure and orientation. Introduction to Computer Vision. Introduction to the quadcopter kit. |
|  | Sep 3 (T) |  | Orientation to Raspberry Pi 3 Computer, Linux Operating System and Python Programming language |
|  | Sep 5 (Th) |  | Quadcopter construction |
|  | Sep 10 (T) |  | Python programs editing and debugging. Math library and string manipulation. |
|  | Sep 12 (Th) | HW1 | Quadcopter construction |
|  | Sep 17 (T) |  | Image formation and representation |
|  | Sep 19 (Th) |  | Quadcopter construction |
|  | Sep 24 (T) |  | Filtering and edge detection |
|  | Sep 26 (Th) | HW2 | Quadcopter construction |
|  | Oct 1 (T) |  | Programming Canny edge detection |
|  | Oct 3 (Th) | Milestone 1 | Flight Tests |
|  | Oct 8 (T) |  | Programming Canny edge detection |
|  | Oct 10 (Th) |  | Hough Lines detection |
|  | Oct 17 (Th) |  | Pi and Pi Cam operational |
|  | Oct 22 (T) |  | Programming Hough Lines detection and manipulation |
|  | Oct 24 (Th) | Milestone 2 | Test flights |
|  | Oct 29 (T) |  | Map construction |
|  | Oct 31 (Th) | HW3 | Feature Detection |
|  | Nov 5 (T) |  | Homogeneous transformation and geometry representation |
|  | Nov 7 (Th) |  | Feature Descriptors |
|  | Nov 12 (T) |  | Feature Descriptors |
|  | Nov 14 (Th) | Milestone 3 | Internals of knnMatch |
|  | Nov 19 (T) |  | Programming matching algorithms |
|  | Nov 21 (Th) |  | Determining transform from match |
|  | Nov 26 (T) | HW4 | Other applications of computer vision |
|  | Dec 3 (T) |  | Other applications of computer vision |
|  | Dec 5 (Th) |  | Demonstrations |
|  | Dec 10 (T) |  | Demonstrations |