

- **Class:**

- **Course Description**

- (From course site) This course provides a thorough grounding in the state of the art in the treatment of digital images, particularly within the context of computer graphics, and digital effects. The course is designed to prepare students to understand existing systems for storage, display, transformation and manipulation of digital images write their own software for working with digital images and undertake creative work and research involving digital images. We cover tools and techniques for the generation, handling and analysis of two-dimensional digital images. Image representation and storage, display, media conversion, painting and drawing, warping, color space operations, enhancement, filtering, and manipulation. Just as digital sound has become the standard for high-quality audio recording, the digital image is becoming the standard form of electronic image. Digital images have the advantages of lossless storage, transmission, and retrieval. Their form greatly facilitates generation, manipulation, and display within a computing environment, and they provide a natural syntax for image representation that pervades the world of computer graphics and visualization. Thus, an understanding of the nature, form, and technology of the digital image is essential to a visualization practitioner. Upon completion of this course, students will know the state of the art in the treatment of digital images in the context of computer graphics, and digital effects. They will understand existing systems for storage, display, transformation and manipulation of digital images, they will be able to write their own software for working with digital images and they will be able to undertake creative work and research involving digital images.

- [Reference Link](#)

- Below are project descriptions for each sub-directory included.

- **Project 01 Rasterization Program**

- **Project Description**

- This program implements a bitmap of different patterns based on command line input response. The program renders different colors or patterns based on input.

- **Language**

- C++

- **Operation**

- Run the command “make -f makefile.mak” to compile the program.

- To run the program, type the command, `./main`

Project 02: Vector to Raster Conversion and Antialiasing

■ Project Description

- This program extends project 01 where a vector description of various shapes are calculated and then rasterized into the bitmap displayed. The program implements anti-aliasing to smooth the edges of the vector-to-raster shapes.

■ Language

- C++

■ Operation

- Run the command `make -f makefile.mak` to compile the program.
- To run the program, type the command, `./pr02`

Project 03: Image Manipulations

■ Project Description

- The goal of this project is to learn global manipulations and use them to create aesthetic results. Most of these operations can be seen under Image/Adjustments in Photoshop such as "Curves, Posterize or Replace Color" operations or in filters. You will also learn how to use parametric functions.

■ Language

- C++

■ Operation

- Run the command `make -f makefile.mak` to compile the program.
- To run the program, type the command, `./pr03 x.ppm y.ppm`
 - X.ppm refers to any .ppm available in the directory.
 - Y.ppm refers to any .ppm available in the directory.

Project 04: Basic Convolution Filters

■ Project Description

- You will implement convolution filters that can be applied to an image file. The goal of this project is to understand filter kernels and their effect on an image. Filters implemented include *motion blur*, *emboss*, *smartblur*, *dilation*, and *erosion*.

■ Language

- C++

■ Operation

- Run the command `make -f makefile.mak` to compile the program.
- To run the program, type the command, `./pr04 <source.ppm> <control.ppm>`

- <source.ppm> refers to any .ppm available in the directory.
- <control.ppm> refers to any .ppm available in the directory.

Project 05: Linear/Affine Transformations and Warping

■ Project Description

● Part 1:

- You will develop an affine transformation tool that uses 3x3 matrices to transform images.
 - Linear Transformations that can be done by 2x2 matrices: Rotation, Scaling, Shear, Mirror
 - Transformations that require 3x3 matrices: Translation and Perspective

● Part 2:

- You will develop a warping tool that provide bilinear and a general transformation.
 - Bilinear warping
 - Any interesting inverse transformation that can be provided by an inverse function

■ Language

- C++

■ Operation

- Run the command “make -f makefile” to compile the program.
- To run the program, type the command, “./pr05 <input.ppm> <output.ppm> <transform>”
 - <source.ppm> refers to any .ppm available in the directory.
 - <control.ppm> refers to any .ppm available in the directory.
 - <transform> includes any typed option from the command line: rotate, translation, scaling, shear, mirror, perspective, bilinear, and custom.

Project 06: Sequential Image Generation for Video

■ Project Description

- The project extends project 05, adding a sequential image processing capability to produce a series of images for video (120 total).

■ Language

- C++

■ Operation

- Run the command “make -f makefile.mak” to compile the program.
- To run the program, type the command, “./pr06 <input.ppm> <output.ppm> <transform>”

- <source.ppm> refers to any .ppm available in the directory.
- <control.ppm> refers to any .ppm available in the directory.
- <transform> includes any typed option from the command line: rotate, translation, scaling, shear, mirror, perspective, bilinear, and custom.