# *3DMM Project*

## Software 3D Graphics Pipeline with Transformation, Lighting, and Texture mapping


***Dr. Ruen-Rone Lee***

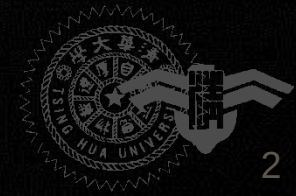
***Computer Graphics and Visualization Lab***

***Computer Science***

***National Tsing Hua University***


# *Outlines*

- Introduction
- Project Objectives
- Project Requirements
- Project Demo

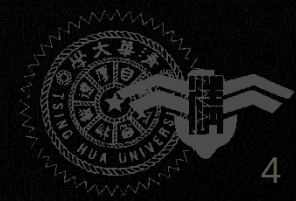


# *Introduction*

- 3D graphics hardware is basically a hardware realization of the 3D graphics pipeline
- Verification and integration of a 3D graphics hardware is a time-consuming task
- Software implementation provide a way for fast functional verification

# *Project Objective*

- Understand more about the details of pipeline algorithms
- Provide a comprehensive knowledge in graphics pipeline processing
- Learn how to debug for a complex graphics rendering process
- Act as a prerequisite for graphics hardware design



# *Project Requirement*

- A software 3D graphics pipeline implementation with following functions
  - Geometry transformation
  - Viewing transformation
  - Lighting manipulation
  - Line/Triangle rasterization
  - Texture mapping

# *Project Requirement*

- **Input**
  - **3D models with texture provided**
- **Output**
  - **A framework that can display the rendering image on a window**
  - **Mouse and keyboard control for transformation, lighting, and rendering mode**

# *Project Requirement*

- **Geometry Transformation**
  - **Translation**
  - **Scaling**
  - **Rotation**
- **Viewing Transformation**
  - **Parallel projection**
  - **Perspective projection**

# *Project Requirement*

- **Lighting Manipulation**
  - **A light source (directional, positional, or spot light) is required to render the scene with different material attributes (ambient, diffuse, specular) for the model**
  - **Multiple light sources are optional**
  - **Phong shading is optional**

# *Project Requirement*

- **Line/Triangle Rasterization**
  - **Line rasterization for wireframe display**
  - **Triangle rasterization for solid display**
  - **Generate attributes for each pixel**
    - **Window coordinates**
    - **Texture coordinates**
    - **Colors**
    - **Depth values**
  - **Depth test for removing hidden pixels**

# *Project Requirement*

- Texture is applied on the input model with texture coordinates provided
- Texture is combined with the color (lighting result) of 3D object by modulation
- Texture filtering to remove texture aliasing