概述

借鉴的官方demo

`Creating a Photogrammetry Command-Line App` https://developer.apple.com/documentation/realitykit/creating_a_photogrammetry_command-line_app

本人demo

本作业自写demo：见 `HW2_demo` 文件夹

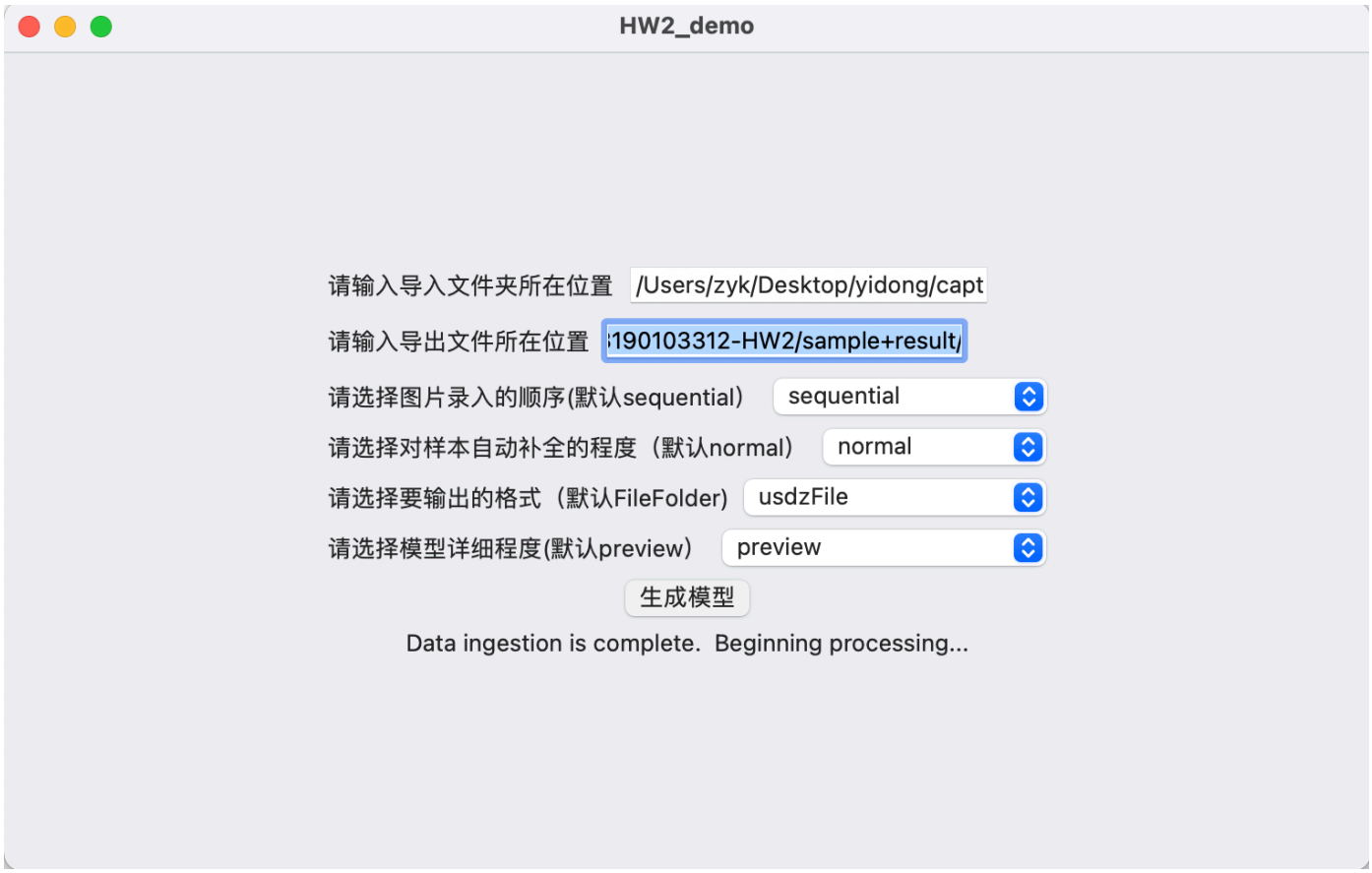
本demo增加/改动部分：

- 可视化界面
- 增加选择直接生成usdz文件的功能
- 实时反馈显示处理进度

界面展示：

可直接将文件夹拖入`import url`和`export url`，注意`import url`中的照片数量在20-200张之间为佳。然后选择好导入顺序、自动补全程度、输出格式、输出模型的详细程度，点击生成模型即可。

模型生成耗费时间较长，一般为3-6分钟左右，因此我在下方设置了一个text实时显示生成进度。



原理简介

使用`PhotogrammetrySession.Request`建立request，并在`makeRequestFromInfo`中获取request的detail信息（输出模型精度）：

```
private func makeRequestFromInfo() -> PhotogrammetrySession.Request {
  if result == "usdzFile"{
    output=output+"/result.usdz"
  }
  let outputUrl = URL(fileURLWithPath: output)
  if let detailSetting = detail{
    return PhotogrammetrySession.Request.modelFile(url: outputUrl, detail:
detailSetting)
  }
  else{
    return PhotogrammetrySession.Request.modelFile(url: outputUrl)
  }
}
```

从`makeConfigurationFromInfo`中获取configuration信息：

```
private func makeConfigurationFromInfo() ->
PhotogrammetrySession.Configuration {
  var configuration = PhotogrammetrySession.Configuration()
  sampleOrdering.map { configuration.sampleOrdering = $0 }
  featureSensitivity.map { configuration.featureSensitivity = $0 }
  return configuration
}
```

使用输入sample和configuration，建立session：

```
var maybeSession: PhotogrammetrySession? = nil
do {
  maybeSession = try PhotogrammetrySession(input:
inputFolderUrl,configuration: configuration)
  ContentView.logger.log("Successfully created session.")
} catch {
  ContentView.logger.error("Error creating session: \(String(describing:
error))")
  Foundation.exit(1)
}
guard let session = maybeSession else {
  Foundation.exit(1)
}
```

开启session，处理request请求：

```
do {
  let request = makeRequestFromInfo()//makerequestinfo从输入中获取信息
  ContentView.logger.log("Using request: \(String(describing: request))")
  try session.process(requests: [ request ])//开启session，处理请求
  RunLoop.main.run()//事实上这里可以循环处理连续的多个请求，但是我这里没有写多个
sample输入的可视化界面
} catch {
  ContentView.logger.critical("Process got error: \(String(describing:
error))")
  Foundation.exit(1)
}
```