



中山大學
SUN YAT-SEN UNIVERSITY

MATLAB 与数字图像处理

Reporter: Quan Zhang
2021/03/09



目录



中山大學
SUN YAT-SEN UNIVERSITY

- **MATLAB介绍**
- **MATLAB编程**
- **实例分享**
- **总结与展望**



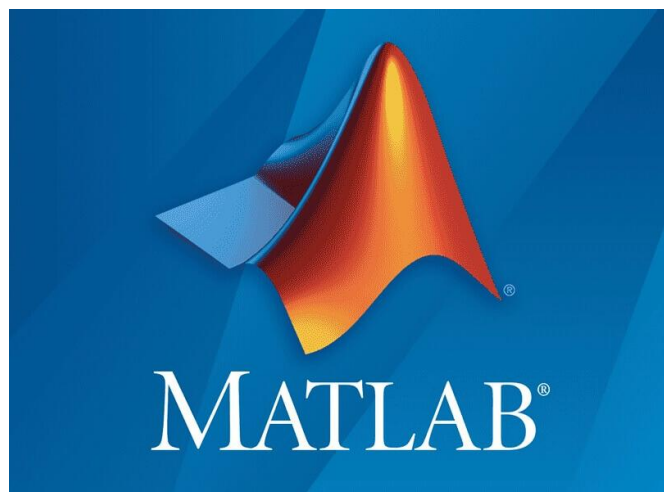
MATLAB介绍



中山大學
SUN YAT-SEN UNIVERSITY

MATLAB, 全称Matrix Laboratory (矩阵实验室)。

MATLAB是一种广泛应用于工程计算、数值分析领域的高级语言，在线性代数、信号分析、图像处理等课程中都有广泛的应用。Matlab功能强大、简单易学、编程效率高，受到广大学者的欢迎，是国际公认优秀的工程应用开发环境。



MATLAB介绍



中山大學
SUN YAT-SEN UNIVERSITY

中山大学向迈斯沃克 (MathWorks) 软件 (北京) 有限公司采购了 MATLAB 校园版软件许可服务, 在校师生可在个人电脑和校属电脑上安装 MATLAB 系列产品, 并享有 MathWorks 服务支持、在线资源和课程开发等服务。

中山大学 | 软件与云服务

首页

校内外购成品软件及云服务发布站

微软产品与服务

福昕PDF编辑器

MATLAB & Simulink

IBM SPSS

更多资源

个人电脑单机版安装指南

适用于师生自带个人电脑 (BYOD)、个人专用的校属资产电脑安装。

1. 注册 MathWorks 帐户及关联许可证

注意1: 仅限于使用 @mail.sysu.edu.cn 和 @mail2.sysu.edu.cn /

注意2: 注册帐号及关联许可证后, 勿直接在 MathWorks 网站下载

2. 下载 MATLAB 安装包 (校内本地下载)

- o R2020b

- o R2020a

- o R2019b

- o R2019a

安装包版本区别

3. 安装并激活 MATLAB 个人电脑单机版 (2020版本)

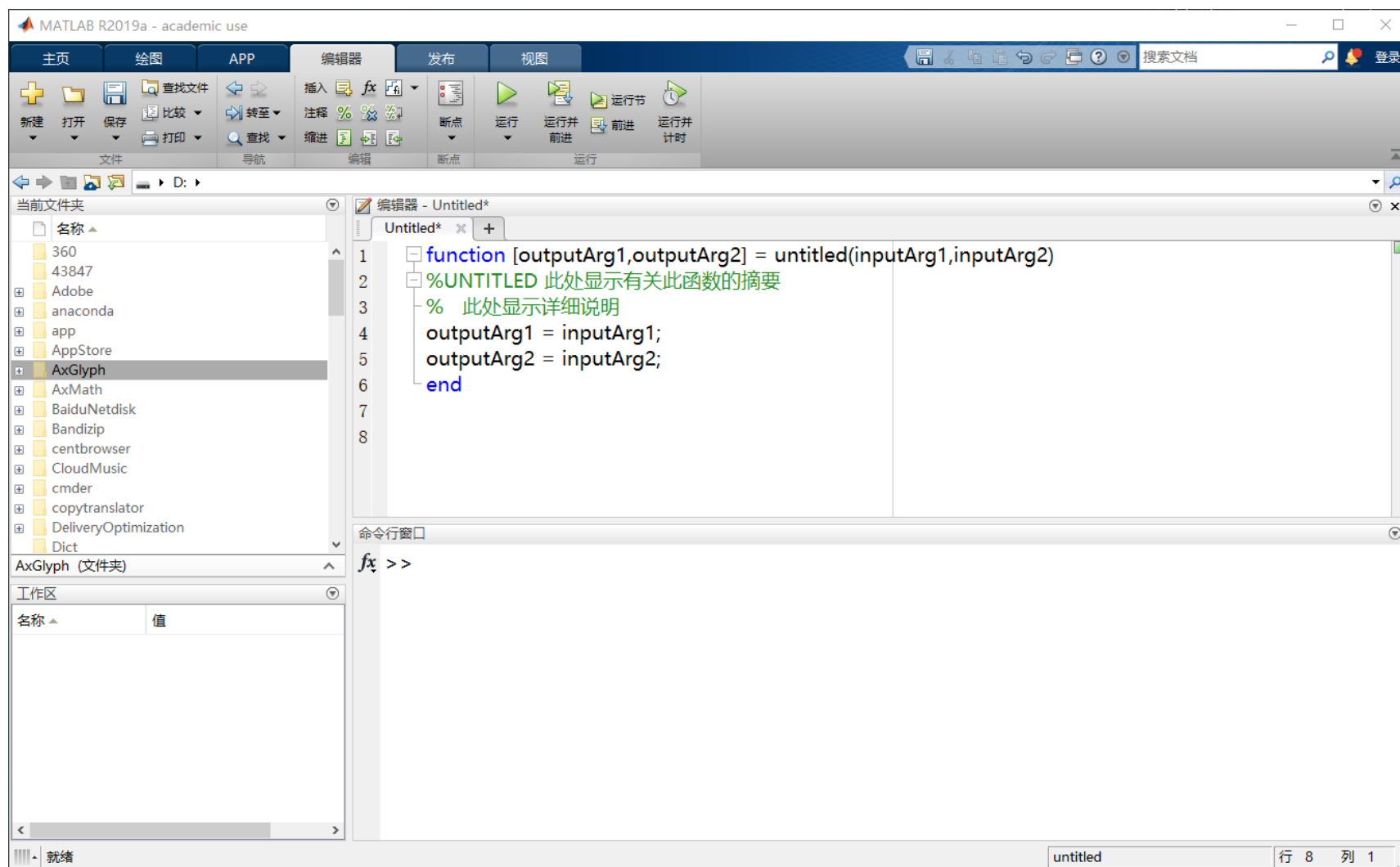
4. 安装并激活 MATLAB 个人电脑单机版 (2019及之前版本)

<https://software.sysu.edu.cn/>

MATLAB介绍



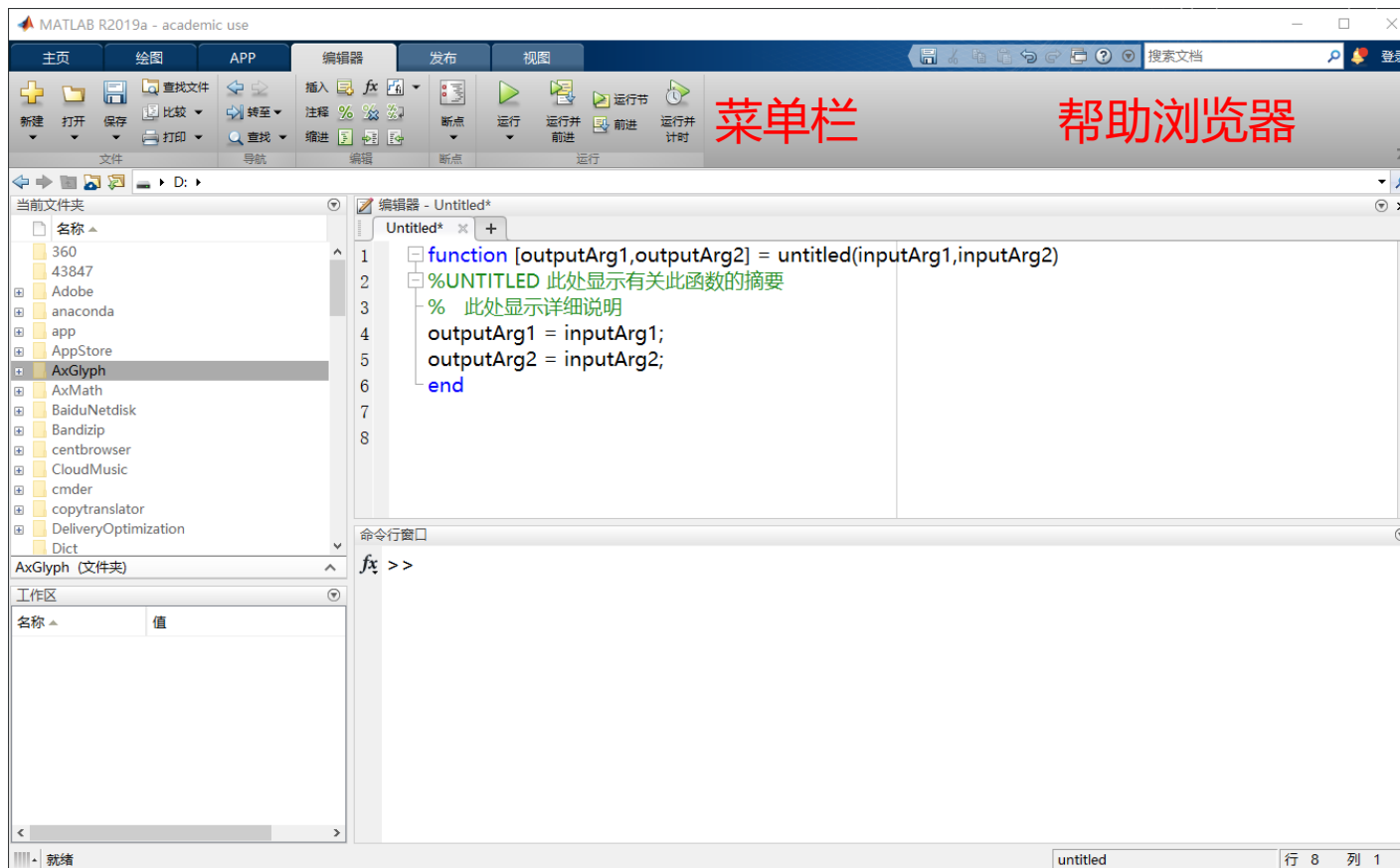
中山大学
SUN YAT-SEN UNIVERSITY



MATLAB介绍



中山大学
SUN YAT-SEN UNIVERSITY



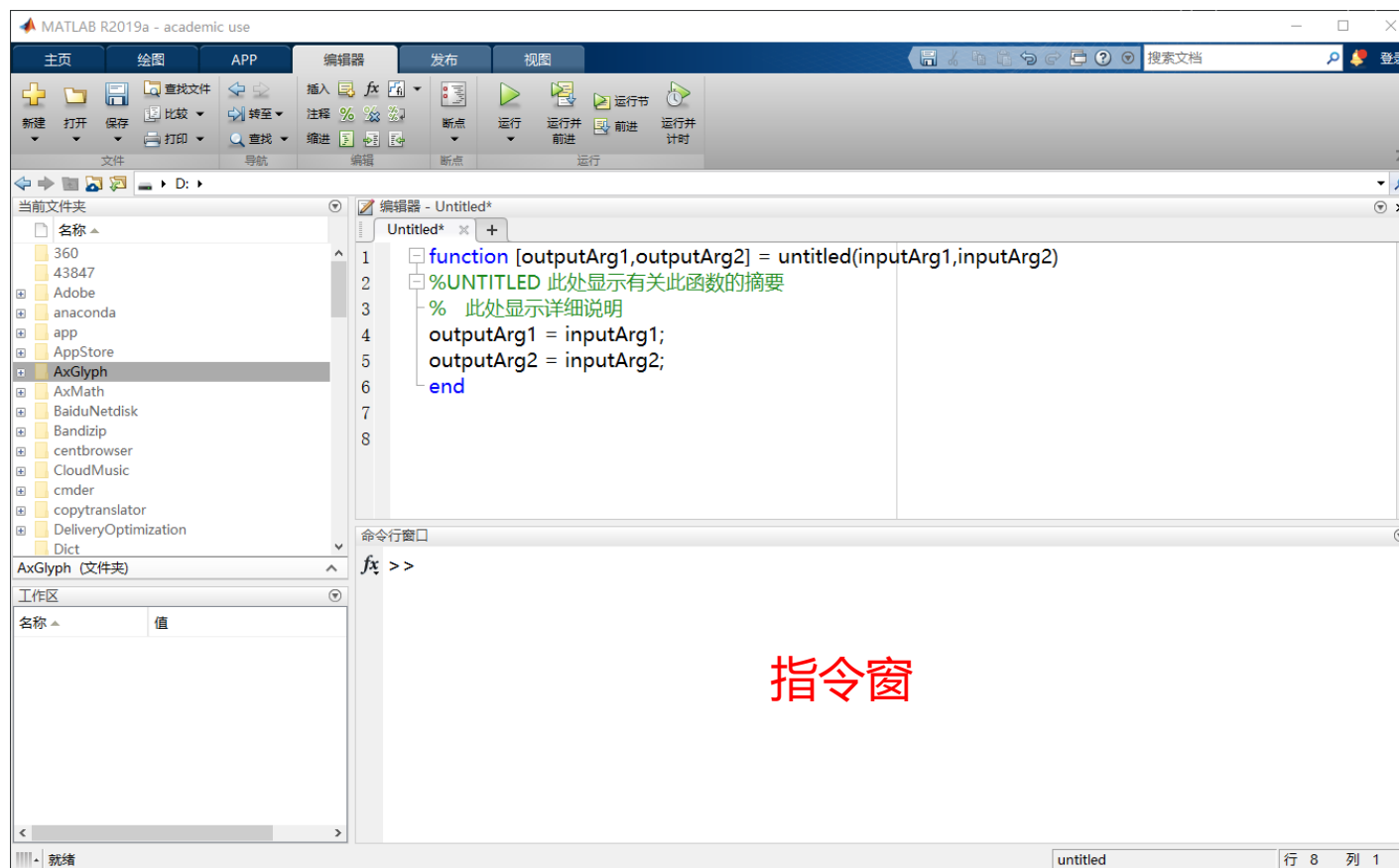
帮助浏览器：该浏览器可以对任何MATLAB中的指令给出详细的理论、算法以及样例，帮助科研人员快速熟悉指令功能。



MATLAB介绍



中山大学
SUN YAT-SEN UNIVERSITY

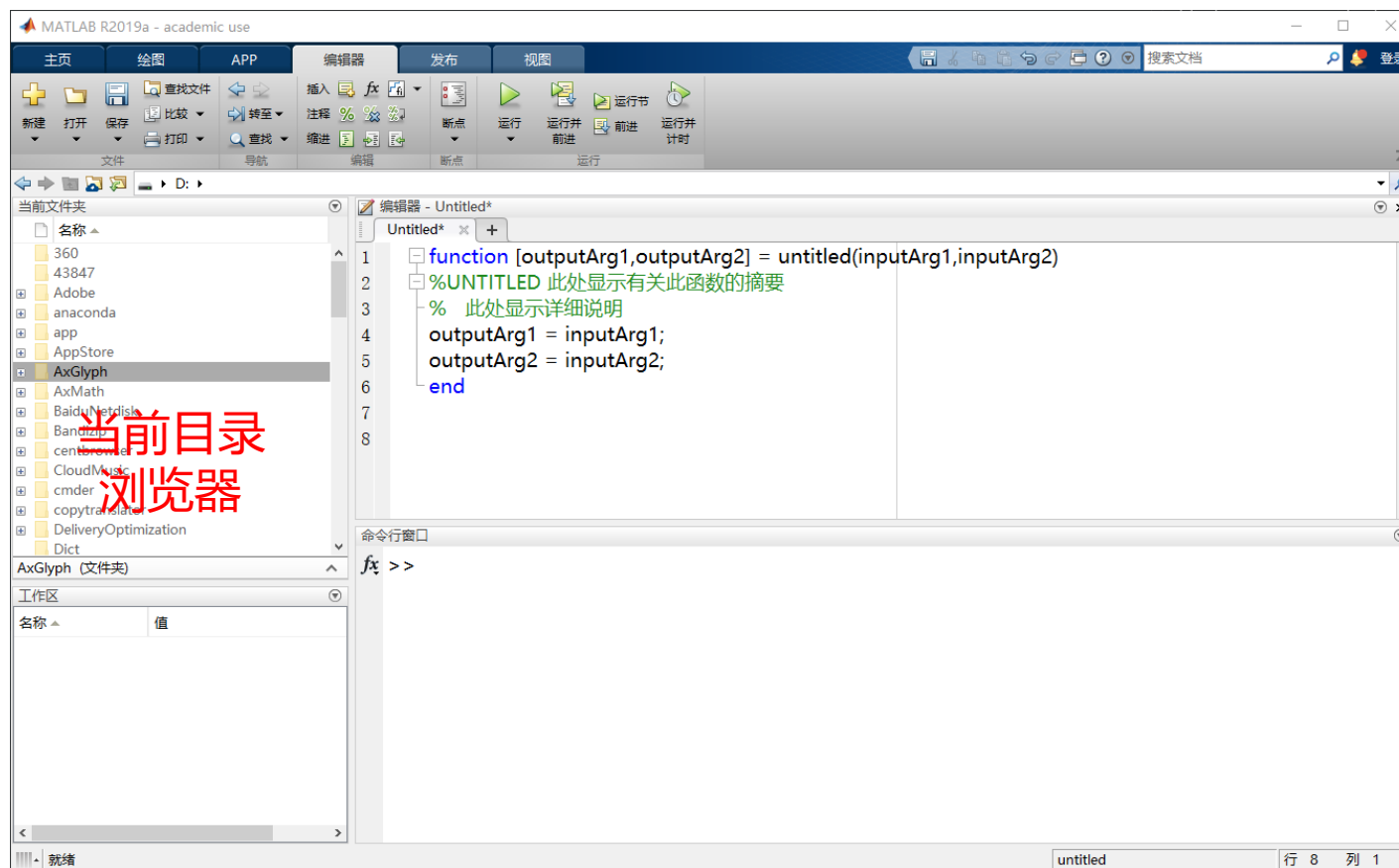


指令窗：该窗是进行各种MATLAB操作的主要窗口。在该窗内，可以键入各种送给MATLAB的指令函数、表达式；显示除图形外的所有运算结果；程序运行错误时，给出相关的出错提示。

MATLAB介绍



中山大学
SUN YAT-SEN UNIVERSITY

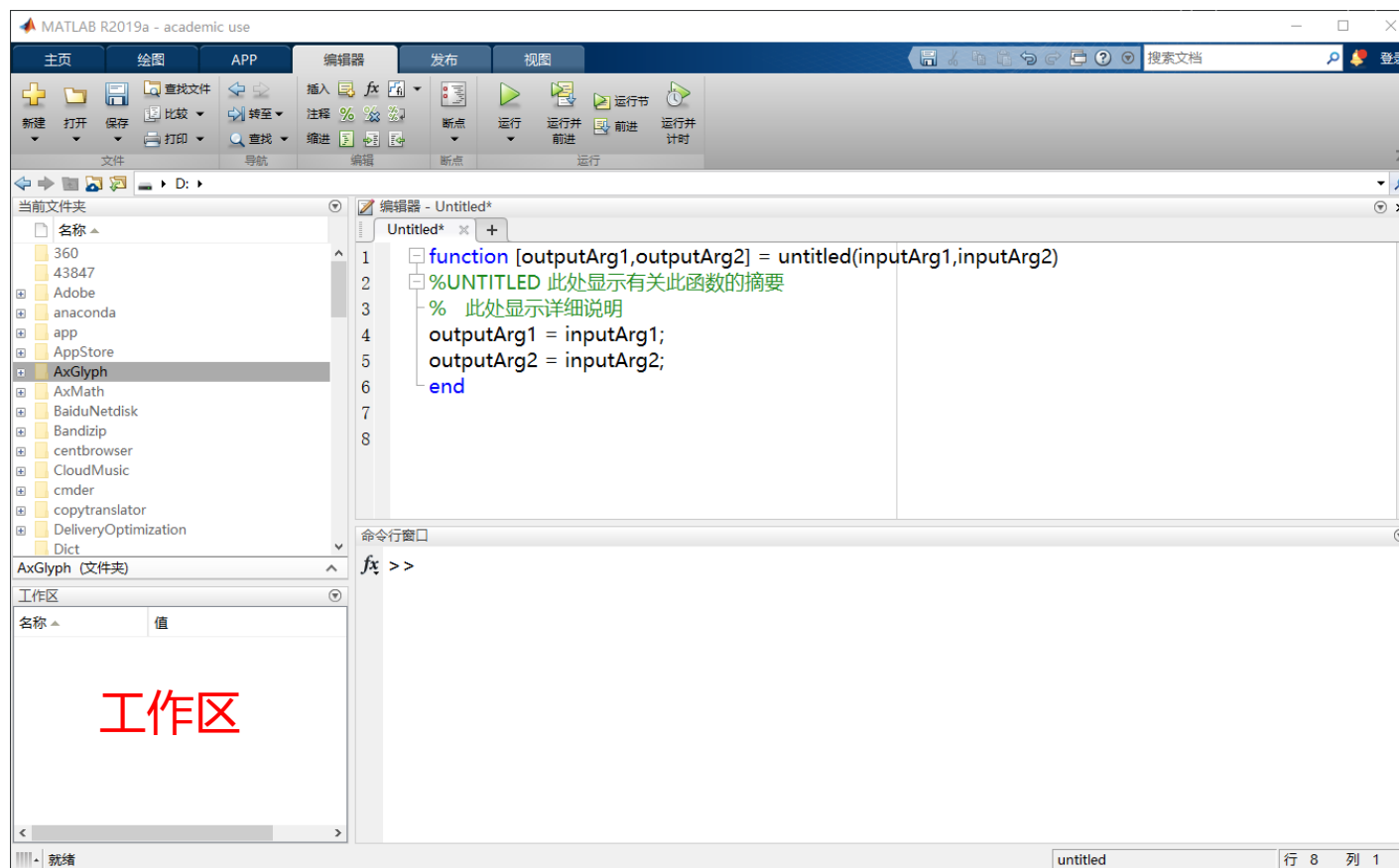


当前目录浏览器：在该浏览器中，展示着当前目录下的所有文件，子文件夹，M文件，MAT文件等，可以直接在该浏览器内进行复制、粘贴、删除。

MATLAB介绍



中山大學
SUN YAT-SEN UNIVERSITY



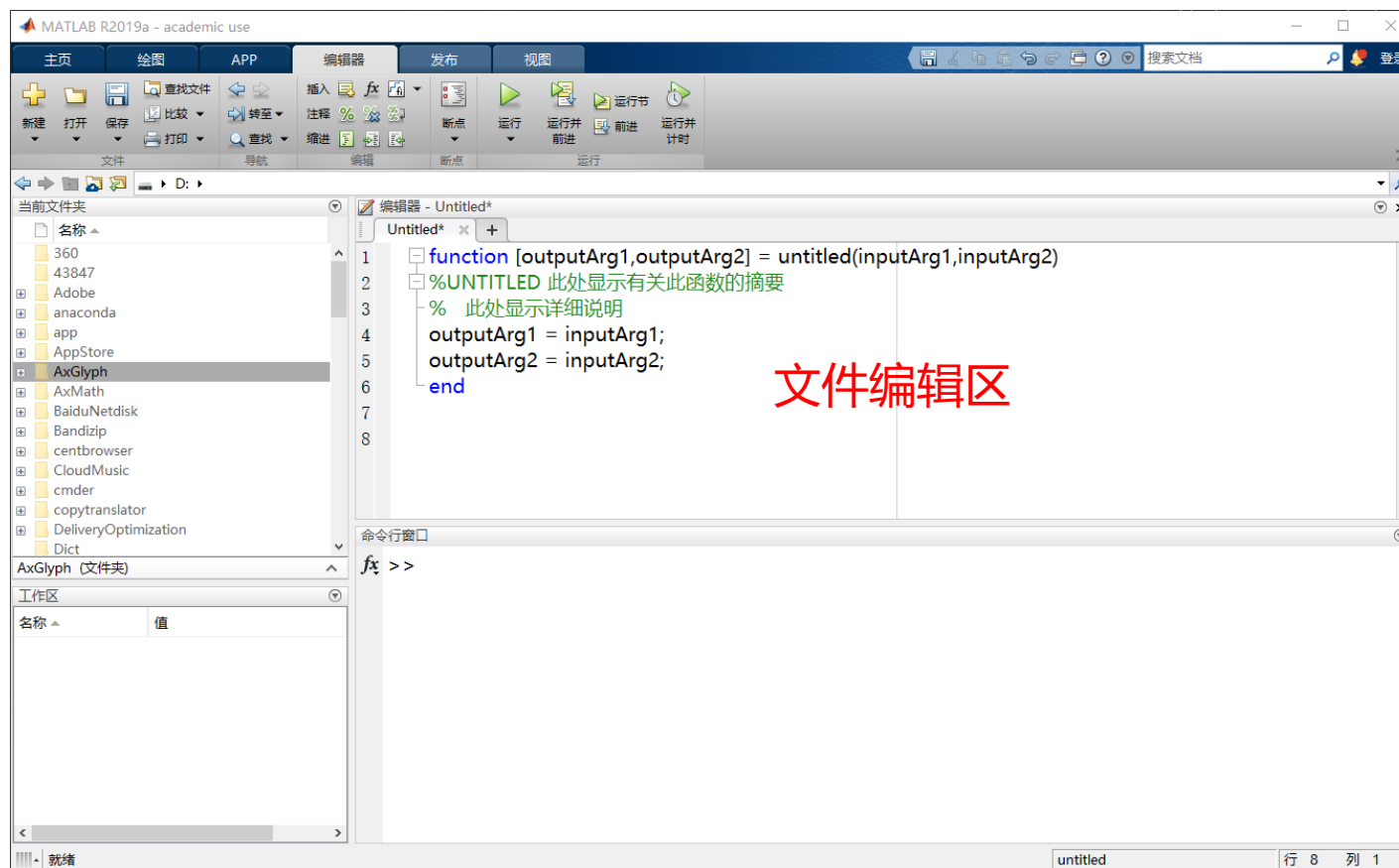
工作区：该窗口罗列出MATLAB工作空间中所有创建过的变量名、大小、取值。可随时进行观察、保存、编辑。



MATLAB介绍



中山大学
SUN YAT-SEN UNIVERSITY



文件编辑区：对所创建的脚本文件或者函数文件进行自定义的编辑。

脚本文件：一般用于要输入的命令较多时，有时需要对其重复输入，可以将这些命令写入扩展名为".m"的文件中，每次运行时直接运行该文件即可。

函数文件：需要被反复多次调用的一个函数，推荐单独写成一个函数文件以供调用。

MATLAB介绍

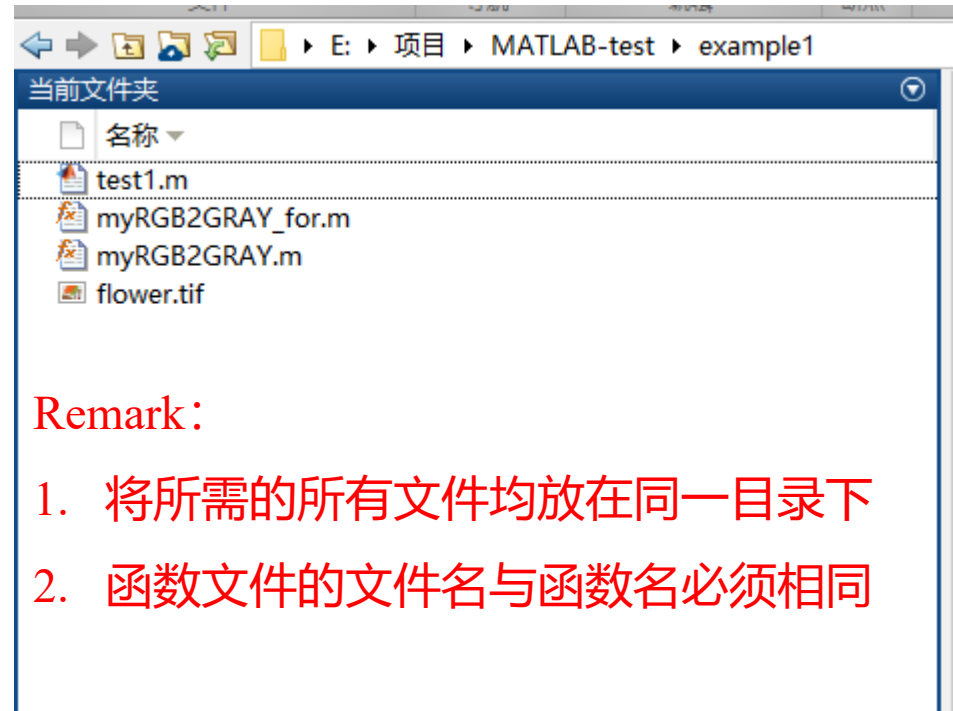


中山大學
SUN YAT-SEN UNIVERSITY

Example: 读入一张彩色图片，将其处理为灰度图片

脚本文件

```
%读取例图
RGB = imread('flower.tif');
%显示例图
imshow(RGB);
%将彩色图片准换成灰度例图
I = myRGB2GRAY(RGB);
%新建一个窗口显示
figure;
%显示灰度图
imshow(uint8(I));
```



函数文件

```
function [imgGray] = myRGB2GRAY(imgRGB)
    imgGray =
    0.299*imgRGB(:, :, 1)+0.587*imgRGB(:, :, 2)+0.11400*imgRGB(:, :, 3);
end
```



MATLAB编程



中山大學
SUN YAT-SEN UNIVERSITY

脚本文件

```
%读取例图
RGB = imread('flower.tif');
%显示例图
imshow(RGB);
%将彩色图片转换成灰度例图
I = myRGB2GRAY(RGB);
%新建一个窗口显示
figure;
%显示灰度图
imshow(uint8(I));
```

在MATLAB中，图像是以多维数组储存在内存中，数组的维度为 $M \times N \times C$ ，其中 M 为图像的行数， N 为图像的列数， C 为通道数，灰度图的通道数为1，RGB彩色图的通道数为3。

工作区	
名称 ▲	值
I	1200x1600 uint8
RGB	1200x1600x3 uint8

函数文件

```
function [imgGray] = myRGB2GRAY(imgRGB)
    imgGray =
    0.299*imgRGB(:,:,1)+0.587*imgRGB(:,:,2)+0.11400*imgRGB(:,:,3);
end
```

MATLAB编程



中山大學
SUN YAT-SEN UNIVERSITY

脚本文件

```
%读取例图
RGB = imread('flower.tif');
%显示例图
imshow(RGB);
%将彩色图片转换成灰度例图
I = myRGB2GRAY(RGB);
%新建一个窗口显示
figure;
%显示灰度图
imshow(uint8(I));
```

Remark:

1. Matlab的变量是不需要预先声明的;
2. Matlab的索引序号是从1开始的;
3. Matlab的句尾分号控制是否打印对应变量。

函数文件

```
function [imgGray] = myRGB2GRAY(imgRGB)
    imgGray =
    0.299*imgRGB(:, :, 1)+0.587*imgRGB(:, :, 2)+0.11400*imgRGB(:, :, 3);
end
```



推荐使用矢量化编程代替For循环操作。

☺ 矢量化编程

```
function [imgGray] = myRGB2GRAY(imgRGB)
    imgGray =
    0.299*imgRGB(:, :, 1)+0.587*imgRGB(:, :, 2)+0.11400*imgRGB(:, :, 3);
end
```

用 “:” 代表某一个维度下的所有元素

☹ For循环

```
function [imgGray] = myRGB2GRAY_for(imgRGB)
    [m,n,~]=size(imgRGB);
    imgGray=zeros(m,n);
    for i=1:m
        for j=1:n
            imgGray(i,j)=
            0.299*imgRGB(i,j,1)+0.587*imgRGB(i,j,2)+0.11400*imgRGB(i,j,3);
        end
    end
end
```



矢量化编程比For循环操作的效率更高。

%读取例图

RGB = imread('flower.tif');

%计算矢量化方法耗时

tic;

I = myRGB2GRAY(RGB);

toc;

%计算for循环方法耗时

tic;

I = myRGB2GRAY_for(RGB);

toc;

>> clear

>> test1_TimeCost

时间已过 0.006659 秒。

时间已过 0.034832 秒。

>> test1_TimeCost

时间已过 0.007152 秒。

时间已过 0.032925 秒。

>> test1_TimeCost

时间已过 0.008165 秒。

时间已过 0.033315 秒。

>> test1_TimeCost

时间已过 0.007059 秒。

时间已过 0.033548 秒。

>> test1_TimeCost

时间已过 0.007373 秒。

时间已过 0.033628 秒。

>> test1_TimeCost

时间已过 0.006690 秒。

时间已过 0.033545 秒。



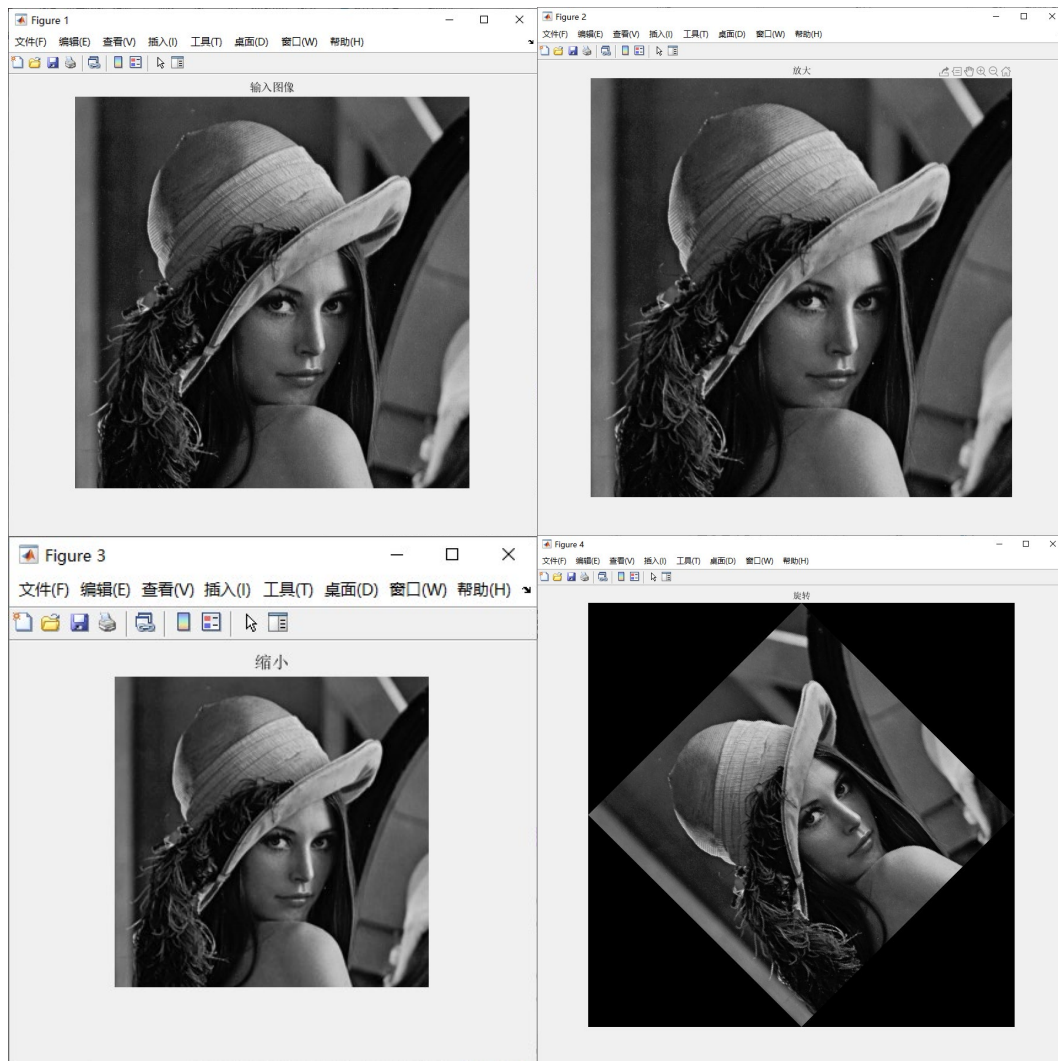
实例分享 (1/2)



中山大學
SUN YAT-SEN UNIVERSITY

Case1: 图像的放大、缩小、旋转

```
clear;  
I=imread('lena_gray.tif');  
im1=imresize(I, 2);  
im2=imresize(I, 0.5);  
im3=imrotate(I,45);  
figure;  
imshow(I),title('输入图像');  
figure;  
imshow(im1),title('放大');  
figure;  
imshow(im2),title('缩小');  
figure;  
imshow(im3),title('旋转');
```



实例分享 (2/2)



中山大學
SUN YAT-SEN UNIVERSITY

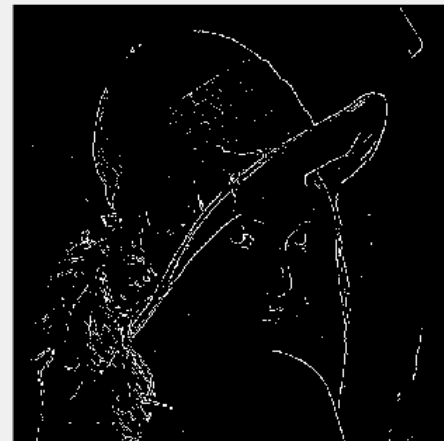
Case2: 图像边缘提取

```
clear;  
I=imread('lena_gray.tif');  
im1=edge(I,'roberts');  
im2=edge(I,'sobel');  
im3=edge(I,'prewitt');  
figure;  
subplot(221);  
imshow(I);  
title('输入图像');  
subplot(222);  
imshow(im1);  
title('roberts');  
subplot(223);  
imshow(im2);  
title('sobel');  
subplot(224);  
imshow(im3);  
title('prewitt');
```

输入图像



roberts



sobel



prewitt

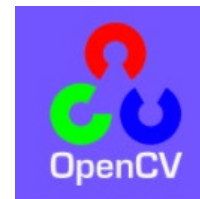
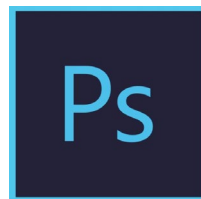


总结与展望



中山大學
SUN YAT-SEN UNIVERSITY

- 对MATLAB的界面，基本操作，编程特色进行了介绍；
- 对使用MATLAB进行图像处理进行了案例分享；
- 鼓励大家多使用MATLAB进行实践，也可进行作业的验证；
- MATLAB也存在局限，鼓励大家也可自学其他框架，提升自己的能力。





中山大學
SUN YAT-SEN UNIVERSITY

Thank you!

