HACETTEPE UNIVERSITY
DEPARTMENT OF GEOMATICS ENGINEERING

–

GMT234
DIGITAL IMAGING & INTERPRETATION
2021-2022

Prof. Dr. Ali Özgün OK

HOMEWORK II REPORT

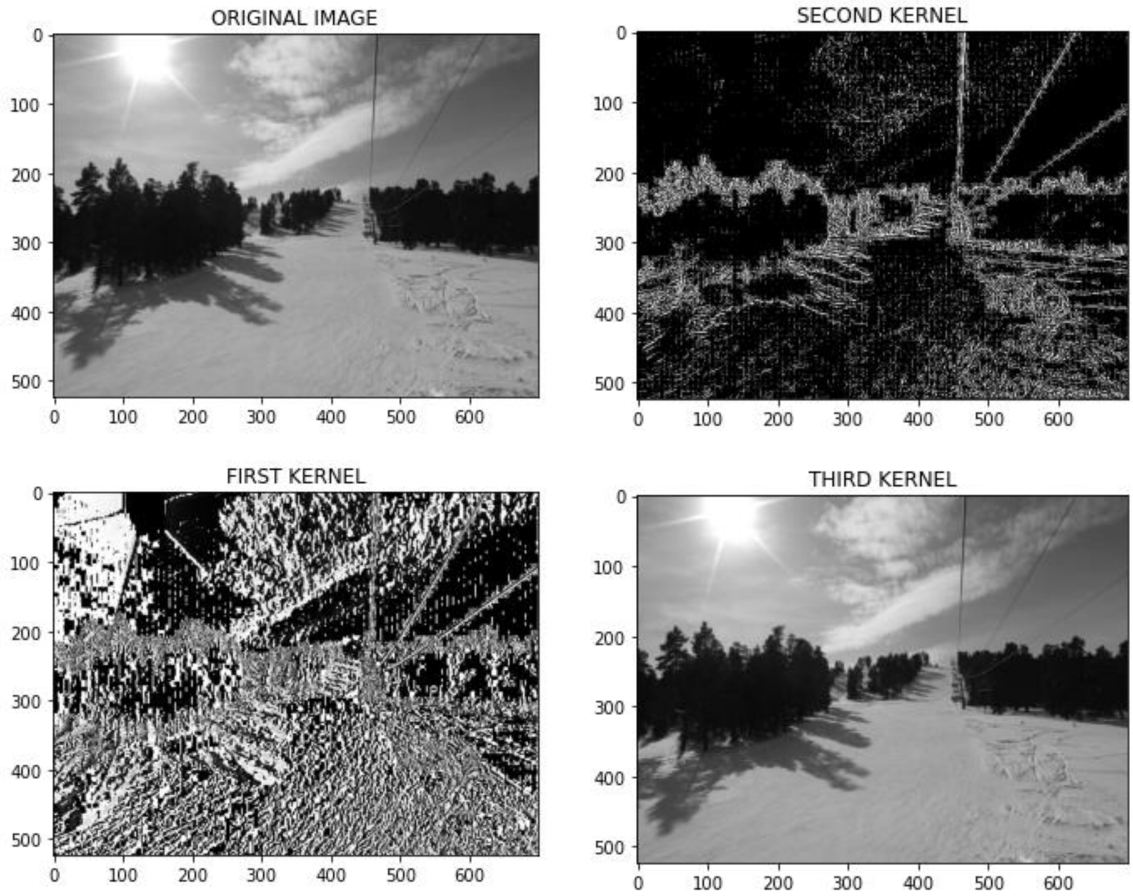
Abdulsamet TOPTAŞ

Deadline	08/05/2022
Delivery date	06/05/2022

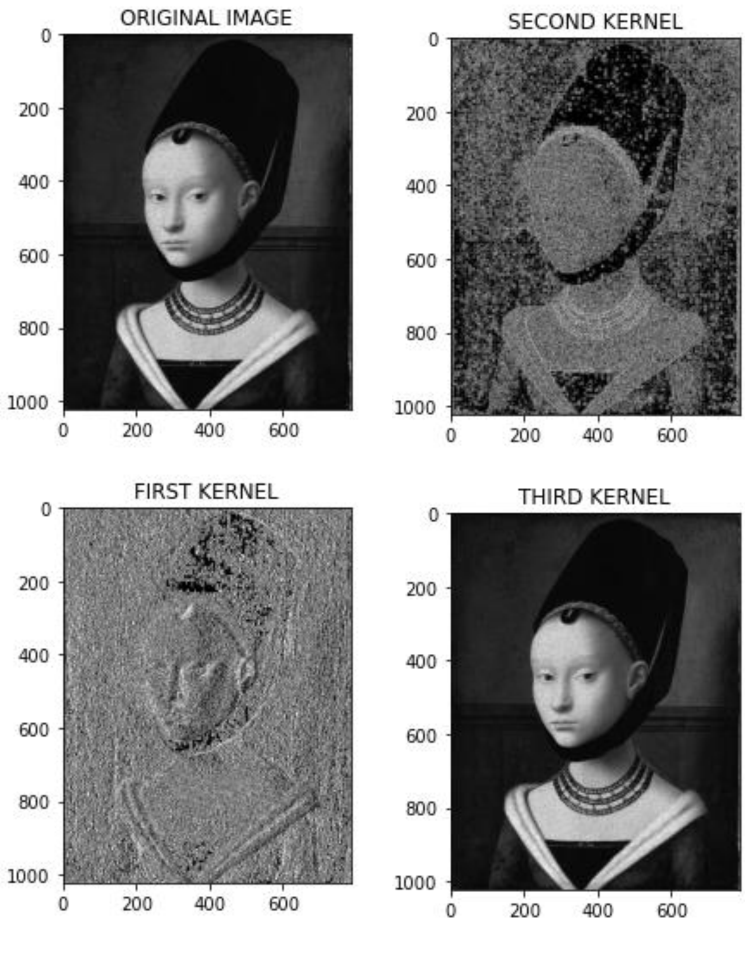
Q1- Define and comment on the output of following kernels. Implement these filters to two different grayscale images you prefer with Python. Provide the outputs (check also Additional Instructions for Report Preparation and Submission given below) (30 points) (must be done without a python library!)

```python
import cv2
import numpy as np
import matplotlib.pyplot as plt

# Read the Image
img = cv2.imread("img1.jpg", cv2.IMREAD_GRAYSCALE)

# Create Colvolution Function
def convolve2d(image, kernel):

    # Convolution output
    output = np.zeros_like(image)

    # Adding zero padding to the input image
    image_padded = np.zeros((image.shape[0] + 2, image.shape[1] + 2))
    image_padded[1:-1, 1:-1] = image

    # Reach all rows and columns of the image
    for x in range(image.shape[1]):
        for y in range(image.shape[0]):
            # Multiplication of kernel and image pixels
            output[y, x] = (kernel * image_padded[y: y+3, x: x+3]).sum()

    return output

# Creating Kernels to be Implemented
First_Kernel = np.array([[1,0,-1],[1,0,-1],[1,0,-1]])
Second_Kernel = np.array([[0,-0.1,0],[-0.2,0.6,-0.2],[0,-0.1,0]])
Third_Kernel = np.array([[1,0,0],[0,0,0],[0,0,0]])

# Show Image and Save Image
plt.imshow(img, cmap='gray')
plt.title('ORIGINAL IMAGE')
plt.show()
cv2.waitKey(0)
cv2.destroyAllWindows()

First_Filter = convolve2d(img,First_Kernel)
plt.imshow(First_Filter, cmap='gray')
cv2.imwrite('FirstKernel_img1.tif',First_Filter)
plt.title('FIRST KERNEL')  # Prewitt X (Left prewitt) Filter
plt.show()
cv2.waitKey(0)
cv2.destroyAllWindows()

Second_Filter = convolve2d(img,Second_Kernel)
plt.imshow(Second_Filter, cmap='gray')
cv2.imwrite('SecondKernel_img1.tif',Second_Filter)
plt.title('SECOND KERNEL')  # Laplacian sharpening Filter
plt.show()
cv2.waitKey(0)
cv2.destroyAllWindows()

Third_Filter = convolve2d(img,Third_Kernel)
plt.imshow(Third_Filter,cmap='gray')
cv2.imwrite('ThirdKernel_img1.tif',Third_Filter)
plt.title('THIRD KERNEL')  # Identity Kernel - the image does not change.
plt.show()
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Output img1;



Output img2 ;



There are 6 new filtered images that I created with Kernel in the file I shared. These images that I save as '.tif' will look clearer and more accurate in the **First_Question file**.

First, I multiplied the kernels given in the first question in Assignment 2 for each 3x3 pixel block in my images, each pixel by the corresponding input of the kernel, and then summed it. In the python code above, I obtained my new images by multiplying and summing the kernels as rows and columns with a 'convolution' function that does these operations. This sum has become the new pixel values in my new image and the filtering process is complete.

I took the "[[1,0,-1],[1,0,-1],[1,0,-1]]" kernel, which I called **FIRST KERNEL** in the output, in my image and analyzed it according to its output. I analyzed this kernel as **Prewitt X (Left Prewitt) Filter**. This kernel or filter is used to detect horizontal edges. **İmportant Note:** In laboratory studies, we have detected a kernel similar to this kernel as **Sobel X Filter**. Although both kernels are similar to each other, according to my research and according to both kernel values and image, I have determined that there is a **Prewitt X Filter**.

I multiplied the "[[0,-0.1.0],[-0.2,0.6,-0.2],[0,-0.1,0]]" kernel, which I named **SECOND KERNEL** in the output, by the pixel values of my image and analyzed it according to the output. I analyzed this kernel as **Laplacian Sharpening Filter**. This kernel or filter is used to highlight edges and fine details in an image. Also, this filter sharpens in all directions.

I analyzed the output by multiplying the "[[1,0,0],[0,0,0],[0,0,0]" kernel, which I named **THIRD KERNEL** in the output, by the pixel values of the image. I analyzed this kernel as **Identity Kernel**. This kernel leaves the image unchanged, so there is no change in pixel values.

Q2- Write the pseudo-code algorithm that automatically counts only the number of objects with hole(s) and the shape of the object is a triangle. Also present a block diagram of your algorithm to make easy to understand your answer. If there any, please also define any assumptions you perform in your algorithm (about the size/shape/tone/hole etc. of the object). (50 points) (No need to implement a python code!)

Pseudocode ;

```
Import cv2
Import numpy as np

# Image reading and determining its shape
Img = Read in grayscale
Row,column = Get the Img shape


# Differentiating with or without holes by specifying threshold values
# So I synced the edges of my holey objects with the background.
For all pixel values in the row
        For the pixels in the [0 and ( column -1 )]
           If equals 255 pixel values in the row and column
               cv2.floodFill(make the columns black (0))
For the pixels in the [0 and (row - 1)]
        For all pixel values in the column
           If equals 255 pixel values in the row and column
               cv2.floodFill(make the rows black (0))

# White Backround
# So my objects became black
cv2.floodFill( make the background of the image white (255) )
```

```
# Print the number of all holey shapes in the image
number of holey objects = 0
    For all pixel values in the row
        For all pixel values in the column
          If equals 0 pixel values in the row and column
            cv2.floodFill( make object without hole (row,column = 255) )
              If the pixels in the row and ( column -1 ) are between 0 and 255
                  number of holey objects  += keep detecting all by adding 1 on objects
                  cv2.floodFill( Set (column -1) and row  to white(255) )
```

print ‘number of holey objects’

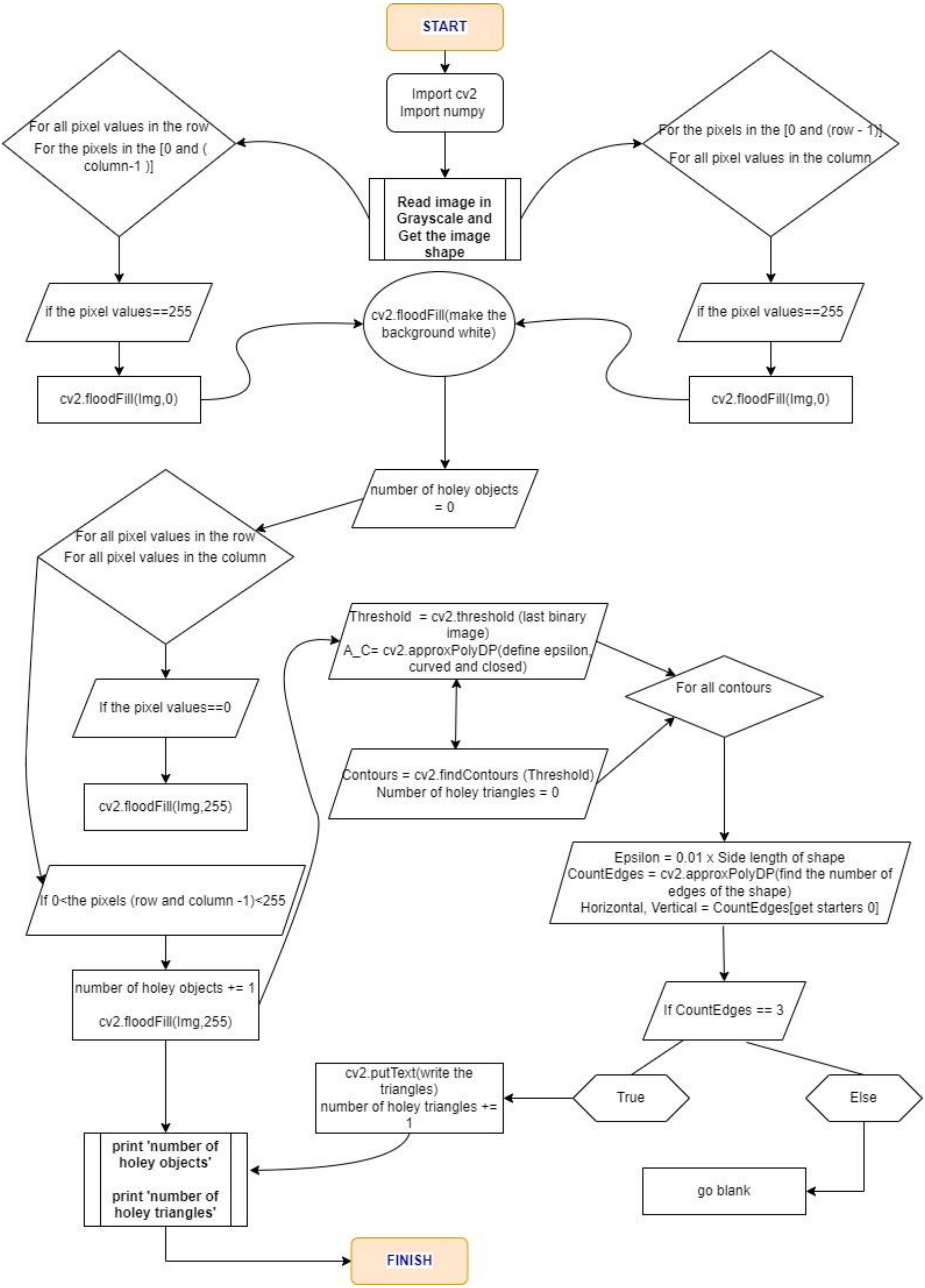
```
# Perforated objects detected
# Now detecting both perforated and triangular shapes
# Shape and size are used for triangle detection

Threshold = cv2.threshold(last binary images)
A_C =cv2.approxPolyDP(define epsilon, curved and closed)
# contours → detect object size and shape
Contours =cv2.findContours(Threshold)
number of holey triangles = 0
For all contours
    Epsilon = 0.01 x Side length of shape
    CountEdges = cv2.approxPolyDP(find the number of edges of the shape)

    Horizontal,Vertical = CountEdges[get starters 0]

    If the length of the CountEdges is equal to 3
        cv2.putText(write the triangles)
        number of holey triangles += keep detecting all by adding 1 on objects
    else
        go blank
```

print ‘number of holey triangles’

Code Comment ;

While doing this part, I benefited from the 'size, shape and hole' features of the object. According to my research, I learned that the 'cv2.floodFill()' algorithm of the OpenCV library will help me in this section. Flood-fill is an algorithm mainly used to determine a bounded area connected to a given node in a multi-dimensional array. Another special command I got help with is cv2.contours() from the OpenCV library, which is a contour finding command. This command helps me to find the contours of the shapes. As a result of this operation, the image contour and hierarchy are revealed. The information assigned to the "contours" variable in the code is actually a python list of the contours in the image. Each contour is a Numpy array of the object's boundary point coordinates (x,y).
While doing Q2, I re-examined the OpenCV library over videos and pdfs.

Block Diagram of Pseudocode ;



Q3- Please investigate and provide information for “augmented reality” applications with your cell phones. Please also discuss at least three “augmented reality” applications that can be utilized in the context of Geomatics Engineering (provide the details for your answer.) (20 points)

Augmented reality is one of the biggest technology trends in the world today. If you have a smartphone like an iPhone or Android, you probably have an app that uses augmented reality (AR). This technology adds digital information to everything you look at from your phone's camera. We now use it in almost every mobile application designed for Geomatics Engineering, and it greatly simplifies the application's fluency and intelligibility. It is actively used in underground facilities, architecture and building designs, and navigation applications in our profession, and it makes our work very easy, I can say that it is used effectively in the applications of the GIS field.

Before moving on to a detailed research, I would like to give a few examples;

- Advanced navigation systems use augmented reality to overlay a route over a live view of the road,
- Neurosurgeons sometimes use AR projection of a 3D brain to assist themselves with surgeries,
- During football games, broadcasters use AR to draw lines on the field to illustrate and analyze the games.

Let's show a navigation with images;

Figure 1 : Augmented reality used

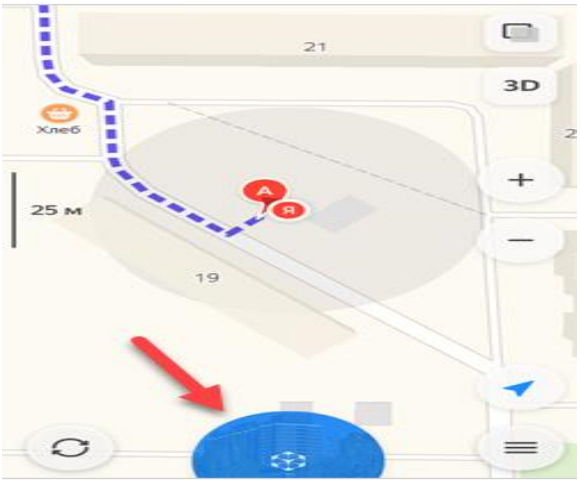


Figure 2 : No Augmented reality used

“ Augmented reality (AR) is an interactive 3D experience that combines a view of the real world with computer-generated elements. ”

Augmented Reality Applications ;

Augmented reality applications are software applications that integrate digital visual content and sometimes audio or similar genres into the real world environment of the user. I will describe some of the best Augmented Reality applications that I have researched on the Internet and then discuss the applications used in Geomatics Engineering.

1. **Houzz**

A great platform for homeware and furniture dealers, Houzz is one of the best AR apps for planning interior layouts and design. Primarily a home improvement app, Houzz has e-

commerce functionality that allows users to browse and purchase in-app products.
The “View In My Room” feature uses AR technology to place products in a photo of the user's home. The resulting image using 3D technology is lifelike. It even goes so far as to show how the product will look in different lighting. Consumers can literally shop for a new sofa from their sofa. Thus, you can have an idea before going to the shopping place, or you can place pinpoint orders for internet ordering.

Figure 3: Houzz 3D App

2. Pokemon Go

Augmented reality has advanced in the entertainment industry as well. One of them is a phone game. I can describe Pokemon Go, one of the first applications that came out during the transition from virtual reality to augmented reality, as a game where you fight and collect Pokemon using your smartphone's camera. Pokemon are mobile and you have to detect them from the phone's camera. It provides these movements using “GPS” location information.

Figure 4: Pokemon Go App

3. Google Lens

One of the leading brands in Augmented Reality is Google. The Google Lens application identifies the contents inside the camera and presents information about them to the screen. This application, which presents many information to its users such as the name of the flower when you show the camera a flower you see while crossing the road, is the latest update of the concept of Google and Google Goggles, which uses the power of the computing cloud to identify the contents.

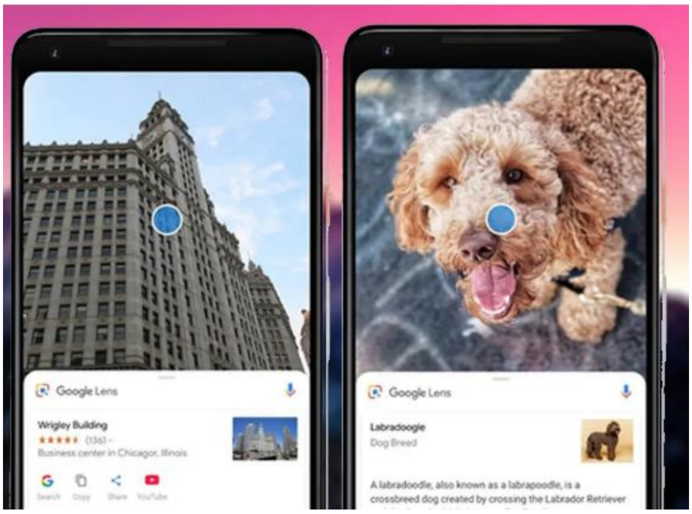


Figure 5: Google Lens App

Augmented Reality Applications Used in Geomatics Engineering ;

1. AuGeo

AuGeo is an Esri Labs initiative to explore the possibilities of using ArcGIS data in an augmented reality environment. Available free to users in the iTunes and Google Play stores, this app allows us to easily bring data from existing ArcGIS point feature layers into an augmented reality application. Anyone can create and enhance a layer in the AuGeo app. This application, which is connected with Geomatics Engineering, we will add a layer on the map with augmented reality, we can navigate in 3D and easily access information.

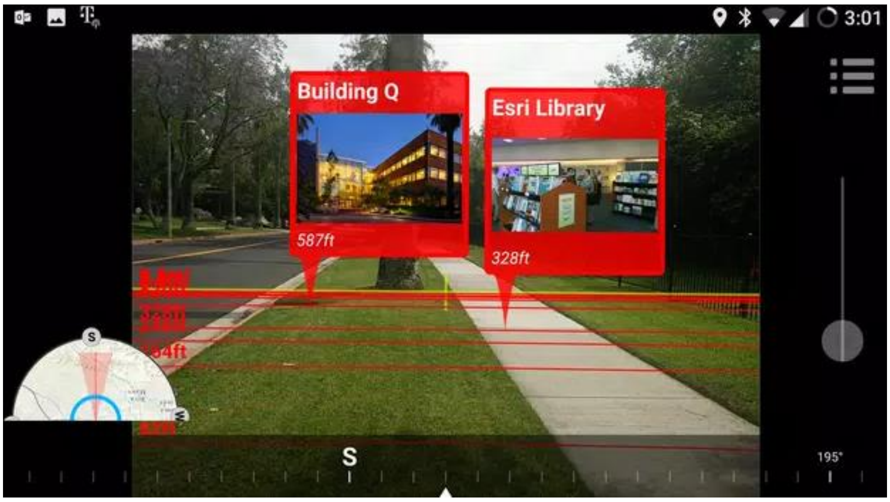


Figure 6: AuGeo Esri App

2. Trimble SiteVision AR

SiteVision opens and simplifies the use of spatial information by allowing you to see your data in the real world, from land boundaries and contours to underground structures and conceptual designs, making it an ideal tool for both the public and contractor in the planning processes and all phases of the project. With the need to acquire and update spatial data accurately and reliably, SiteVision helps create efficient outputs and make good decisions based on the best available information. It is an application suitable for augmented reality, where we can place a 3D model on the image with 2cm sensitivity over the phone, visualize the projects that are not seen underground and projects that have not yet been implemented on the surface, and visualize the plan we set up in our minds for all kinds of maps and architectural projects in 3D on the application.

Figure 7: Trimble SiteVision App

3. vGIS

vGIS accepts traditional BIM and GIS files (IFC, DWG, DXF and others) or dynamically pulls data from Autodesk Build (BIM 360), Bentley iTwin, Esri ArcGIS and other repositories accurately Augmented Reality (AR) of the surrounding infrastructure at the project site) creates views. When combined with common platforms like Autodesk Build and Esri ArcGIS, vGIS automatically incorporates all changes in the AR view as your data changes in the source. Viewing current and proposed models as 3D objects in AR gives you a better understanding of your environment, increases productivity, helps validate accuracy, reduces errors, and provides a safer work environment. In addition, vGIS has advantages such as centimeter-precision visuals, automated data workflows with Autodesk, Bentley and Esri platforms, broad compatibility with off-the-shelf commercial hardware, complete productivity workflows, data collection and 3D scanning tools, and remote assistance capabilities.

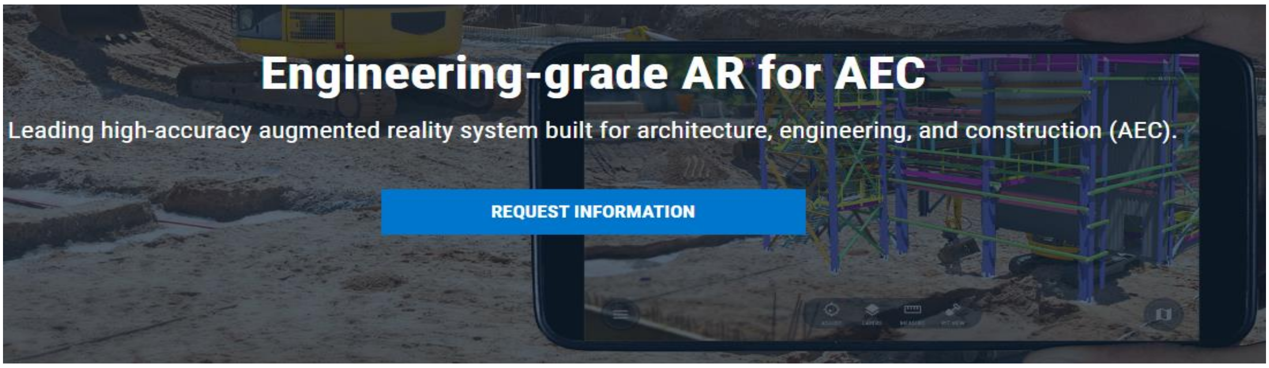


Figure 8: vGIS


Figure 9: vGIS App

4. Google Maps AR

Google Maps AR is an app designed to let you use augmented reality to help you navigate while walking. Rather than just presenting a map, it uses the camera on the back of the phone to determine where you are, superimposing the direction and details on the screen. GPS helps position you on the map, while describing what you can see. As a result, Google Maps AR helps to solve the problem of not knowing which way to walk, because AR can show the right direction, making it easy for you to navigate in the right direction. In Google Maps AR, you will be prompted to lift your phone to use the augmented reality functions. If you hold your phone straight you won't be able to see that view, but you can force it to appear by pressing the pointer icon in the lower left corner. Google Maps tries to identify what's around you by scanning what your camera sees and comparing it to Street View images. This takes place at the top of the screen, while at the bottom of the screen are normal walking directions. Google Maps Live View shows road names and distances in AR as well as major turns to keep you on the right track.

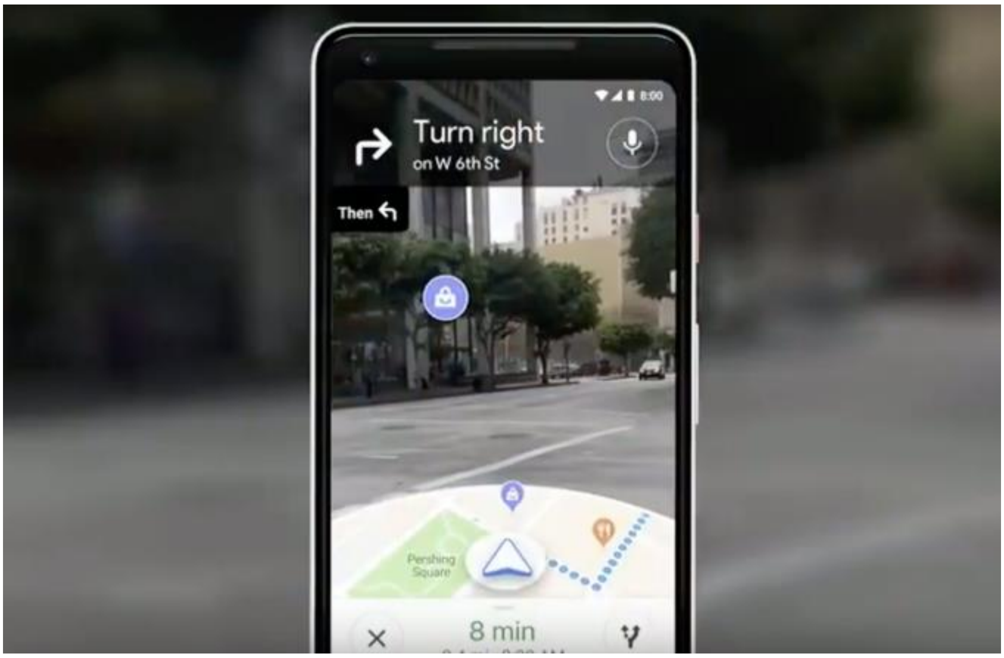


Figure 10: Google Maps AR App