In [1]:

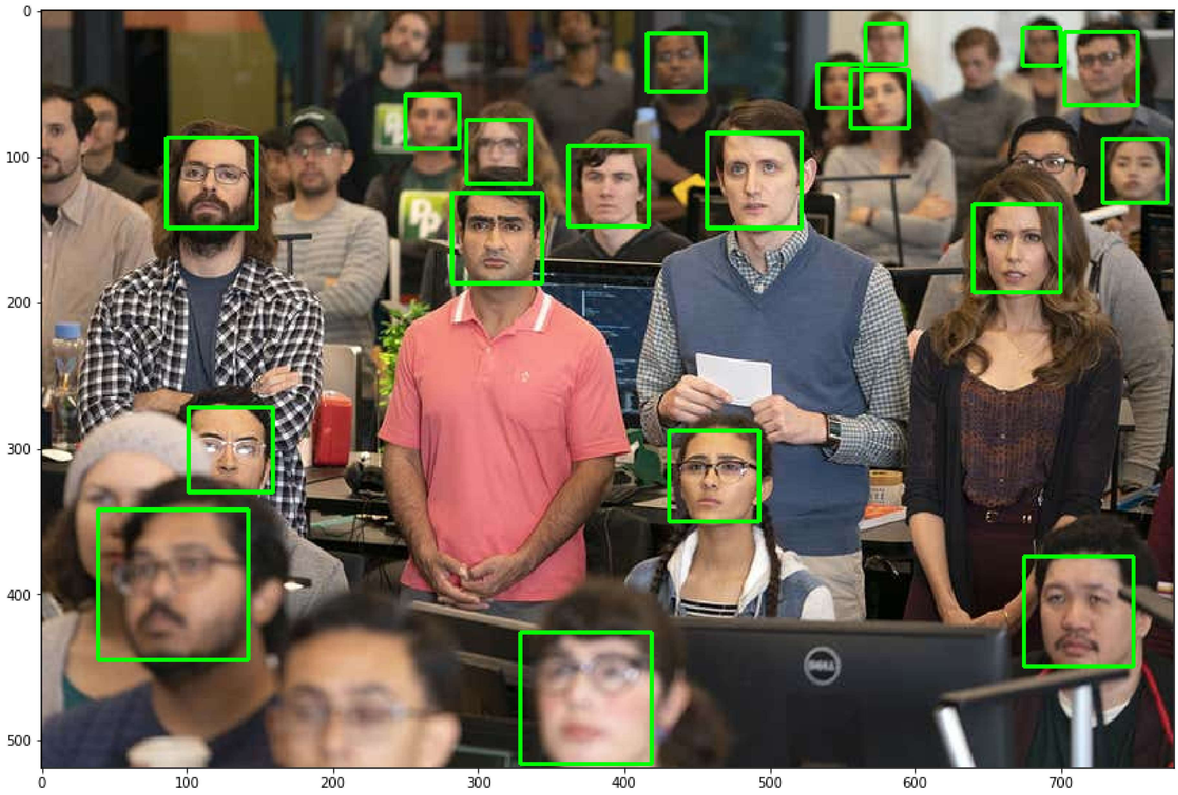
```
import cv2
import numpy as np
import matplotlib.pyplot as plt
```

In [3]:

```
imagePath = './data/faces2.jpg'
cascPath = './xml/haarcascade_frontalface_default.xml'

# Create the haar cascade
faceCascade = cv2.CascadeClassifier(cascPath)

# Read the image
image = cv2.imread(imagePath)
imageRGB = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

# Detect faces in the image
faces = faceCascade.detectMultiScale(gray,
    minNeighbors=5,    # 얼굴 사이의 최소 간격(픽셀)입니다
    minSize=(20, 20),   # 얼굴의 최소 크기입니다
)

# 검출된 얼굴 주변에 사각형 그리기
for (x, y, w, h) in faces:
    cv2.rectangle(imageRGB, (x, y), (x+w, y+h), (0, 255, 0), 2)

# 얼굴을 검출한 이미지를 화면에 띄웁니다
plt.figure(figsize=(15, 15))
plt.imshow(imageRGB)
```

Out[3]:

```
<matplotlib.image.AxesImage at 0x28035900308>
```



In [5]:

```
def FaceDetection(image):
    faceimage = image.copy()

    # Create the haar cascade
    faceCascade = cv2.CascadeClassifier(cascPath)

    # Read the image
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

    # Detect faces in the image
    faces = faceCascade.detectMultiScale(gray, minNeighbors=5, minSize=(20, 20))

    # 검출된 얼굴 주변에 사각형 그리기
    for (x, y, w, h) in faces:
        cv2.rectangle(faceimage, (x, y), (x+w, y+h), (0, 255, 0), 2)

    return faceimage
```

In [7]:

```
cap = cv2.VideoCapture('./data/siliconvalley.mp4')
frame_size = (int(cap.get(cv2.CAP_PROP_FRAME_WIDTH)), int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT)))
print('frame_size =', frame_size)

while True:
    retval, frame = cap.read() # 프레임 캡처
    if not retval:
        break

    face = FaceDetection(frame)
    cv2.imshow('face',face)

    key = cv2.waitKey(25)
    if key == 27: # Esc
        break

if cap.isOpened():
    cap.release()

cv2.destroyAllWindows()
```

```
frame_size = (576, 324)
```

In [ ]: