

- ✓ OpenCV Tutorial
- ☐

OpenCV Tutorial
- ☐

OpenCV Installation
- ☐

Read & Save Images
- ☐

Basic Operation On Images
- ☐

OpenCV Resize Image
- ☐

OpenCV Image Rotation
- ☐

OpenCV Drawing Functions
- ☐

OpenCV Blob Detection
- ☐

Canny Edge Detection
- ☐

OpenCV Gaussian Blur
- ☐

OpenCV Image Filters
- ☐

OpenCV Image Threshold
- ☐

OpenCV Contours
- ☐

OpenCV Mouse Event
- ☒

OpenCV Template Matching
- ☐

OpenCV Erosion & Dilation
- ☐

OpenCV Video Capture
- ☐

Face Recognition & Face Detection
- ☐

Limitations in Face Detection

## OpenCV Template Matching

Template matching is a technique that is used to find the location of template images in a larger image. OpenCV provides the **cv2.matchTemplates()** function for this purpose. It simply slides the template images over the input image and compares the templates and patch under the input image.

There are various methods available for the comparison; we will discuss a few popular methods in further topics.

It returns a grayscale image, where every pixel represents the number of the neighborhood of that pixel match with the input templates.

### Template matching in OpenCV

The templates matching consist of the following step:

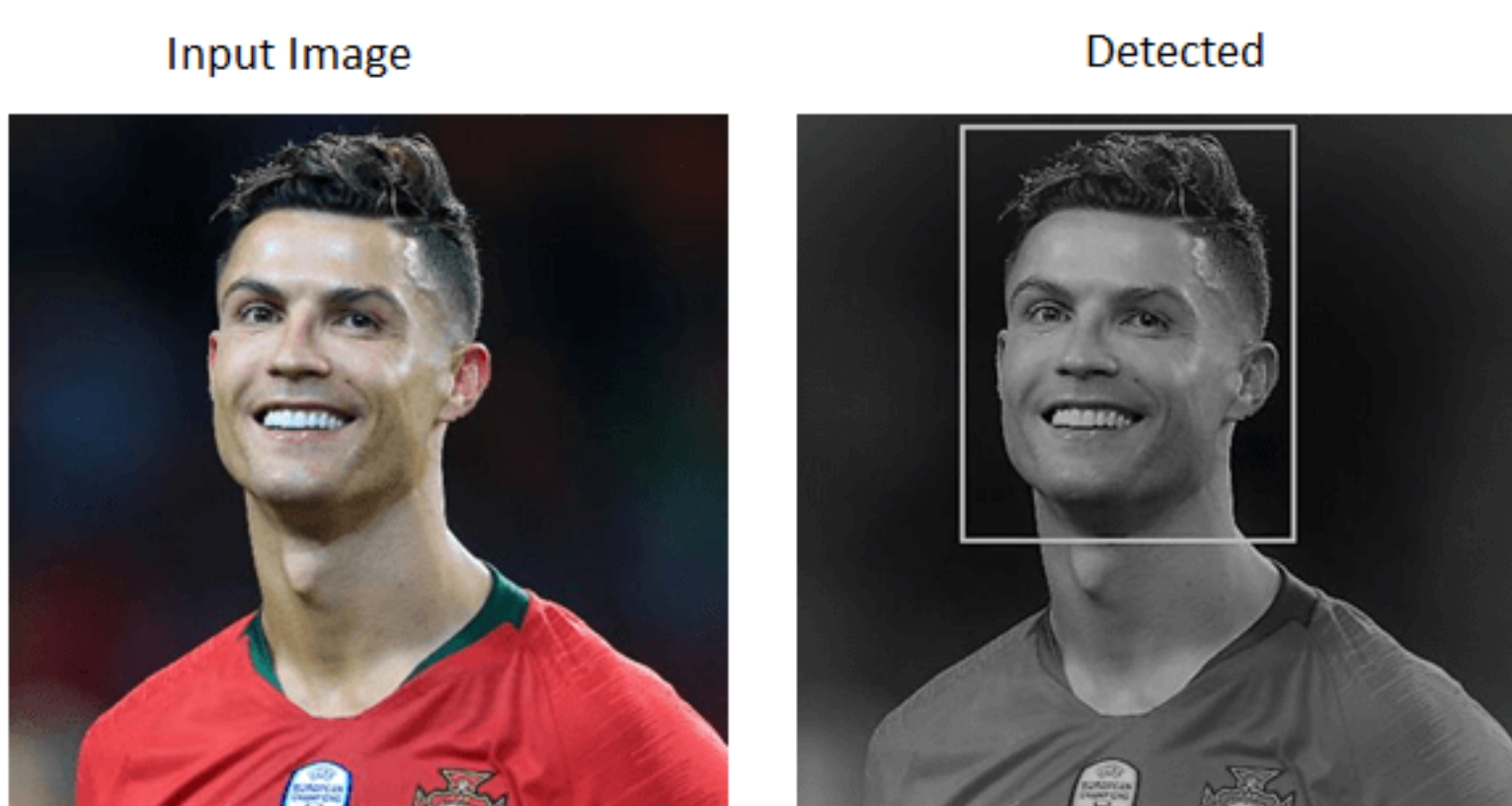
- Step - 1:** Take the actual image and convert it into a grayscale image.
- Step - 2:** Select the template as a grayscale image.
- Step - 3:** Find the location where the accuracy level matches. It is done by template image slide over the actual image.
- Step - 4:** When the result is greater than the accuracy level, mark that position as detected.

Consider the following example:

```
import cv2
import numpy as np

# Reading the main image
rgb_img = cv2.imread(r'C:\Users\DEVANSH SHARMA\rolando.jpg',1)
# It is need to be convert it to grayscale
gray_img = cv2.cvtColor(rgb_img, cv2.COLOR_BGR2GRAY)
# Reading the template image
template = cv2.imread(r'C:\Users\DEVANSH SHARMA\ronaldo_face.jpg',0)
# Store width in variable w and height in variable h of template
w, h = template.shape[::-1]
# Now we perform match operations.
res = cv2.matchTemplate(gray_img,template,cv2.TM_CCOEFF_NORMED)
# Declare a threshold
threshold = 0.8
# Store the coordinates of matched location in a numpy array
loc = np.where(res >= threshold)
# Draw the rectangle around the matched region.
for pt in zip(*loc[::-1]):
    cv2.rectangle(img_rgb, pt, (pt[0] + w, pt[1] + h), (0,255,255), 2)
# Now display the final matched template image
cv2.imshow('Detected',img_rgb)
```

Output:



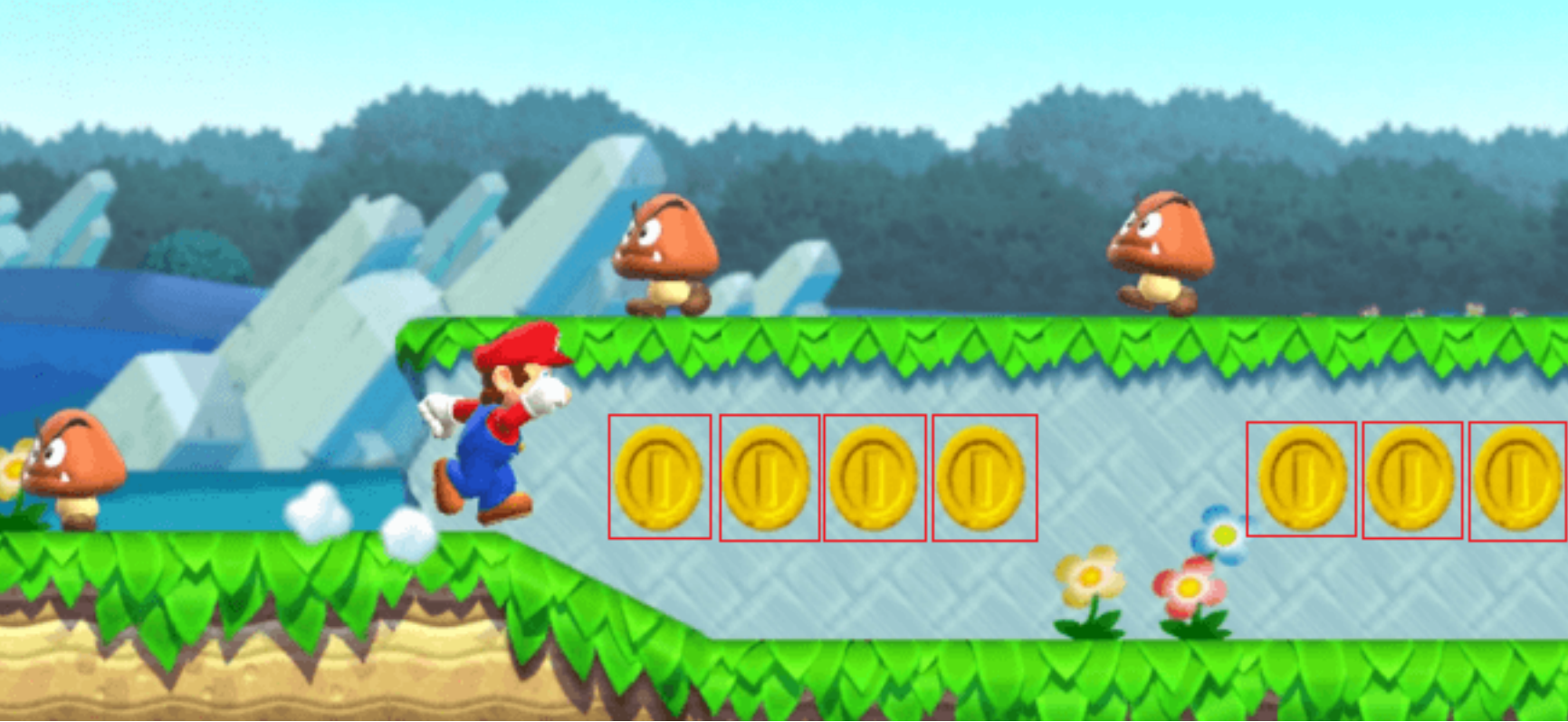
### Template Matching with Multiple Objects

In the above example, we searched image for template image that occurred only once in the image. Suppose a particular object that occur multiple times in particular image. In this scenario, we will use the thresholding because **cv2.minMaxLoc()** won't give all location of template image. Consider the following example.

```
import cv2
import numpy as np

# Reading the main image
img_rgb = cv2.imread(r'C:\Users\DEVANSH SHARMA\mario.png',1)
# It is need to be convert it to grayscale
img_gray = cv2.cvtColor(img_rgb, cv2.COLOR_BGR2GRAY)
# Read the template
template = cv2.imread(r'C:\Users\DEVANSH SHARMA\coin1.png',0)
# Store width in variable w and height in variable h of template
w, h = template.shape[::-1]
# Now we perform match operations.
res = cv2.matchTemplate(img_gray,template,cv2.TM_CCOEFF_NORMED)
# Declare a threshold
threshold = 0.8
# Store the coordinates of matched region in a numpy array
loc = np.where( res >= threshold)
# Draw a rectangle around the matched region.
for pt in zip(*loc[::-1]):
    cv2.rectangle(img_rgb, pt, (pt[0] + w, pt[1] + h), (0,255,255), 2)
# Now display the final matched template image
cv2.imshow('Detected',img_rgb)
```

Output:



In the above program, we took an image of popular super Mario game as main image and coin image as template image. The coins occur multiple times in main image. When it find the coin in the image it draw rectangle on the coin.

### Limitation of Templates Matching

There are few limitations in template matching given as follows:

- It is a time-consuming process to calculate the pattern correlation image for medium to large images.
- Pattern occurrence has to preserve the orientation of the reference template image.a
- Template matching doesn't apply on the rotated or scaled version of the template as a change in shape/size/shear etc.

### Help Others, Please Share



Join Javatpoint Test Series

Placement Papers

TCS

HCL

Infosys

IBM

Accenture

AMCAT

eLitmas

Java

Python

C Programming

Networking

Bank PO/Clerk

UPSSSC

Government Exams

SSC

Civil Services

SBI

GATE

NEET

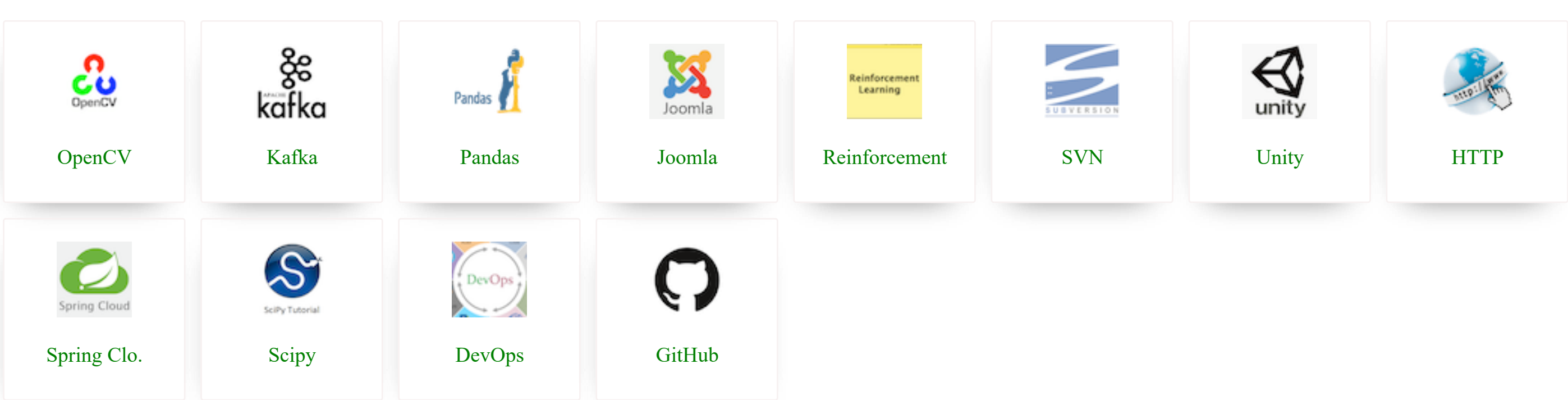
CAT

Railway

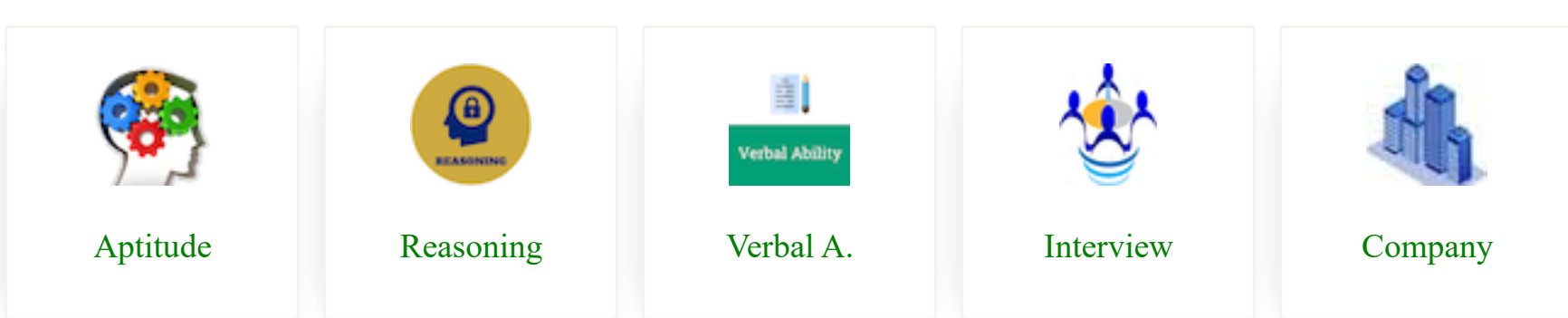
CTET

IIT JEE

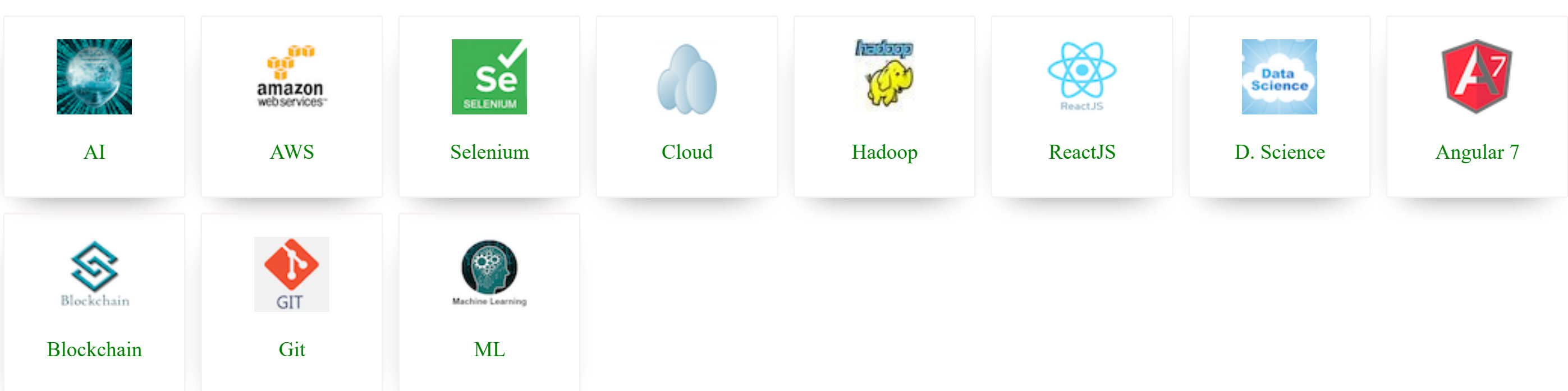
### Learn Latest Tutorials



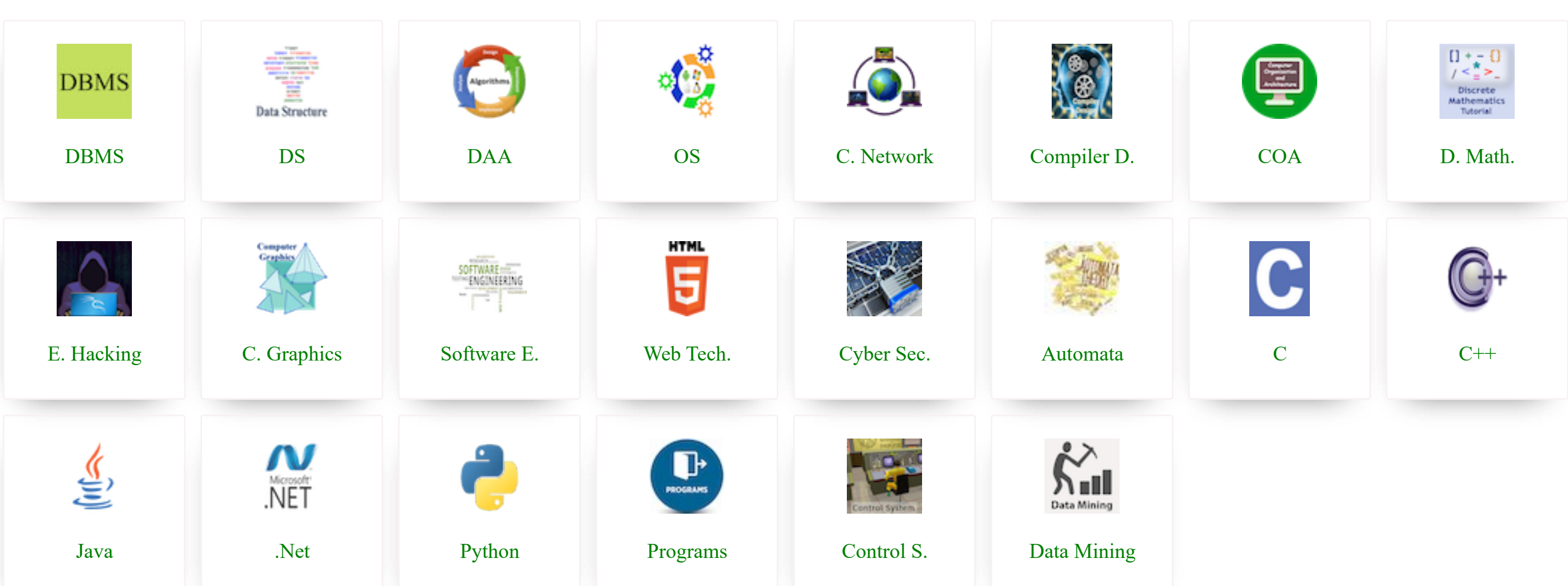
### Preparation



### Trending Technologies



### B.Tech / MCA



### Javatpoint Services

JavaTpoint offers too many high quality services. Mail us on [hr@javatpoint.com](mailto:hr@javatpoint.com), to get more information about given services.

- Website Designing
- Website Development
- Java Development
- PHP Development
- WordPress
- Graphic Designing
- Logo
- Digital Marketing
- On Page and Off Page SEO
- PPC
- Content Development
- Corporate Training
- Classroom and Online Training
- Data Entry

### Training For College Campus

JavaTpoint offers college campus training on Core Java, Advance Java, .Net, Android, Hadoop, PHP, Web Technology and Python. Please mail your requirement at [hr@javatpoint.com](mailto:hr@javatpoint.com).

Duration: 1 week to 2 week

Like/Subscribe us for latest updates or newsletter



#### LEARN TUTORIALS

Learn Java  
Learn Data Structures  
Learn C Programming  
Learn C++ Tutorial  
Learn C# Tutorial  
Learn PHP Tutorial  
Learn HTML Tutorial  
Learn JavaScript Tutorial  
Learn jQuery Tutorial  
Learn Spring Tutorial

#### OUR WEBSITES

Javatpoint.com  
Hindi100.com  
Lyriesia.com  
Quoteperson.com  
Jobandplacement.com

#### OUR SERVICES

Website Development  
Android Development  
Website Designing  
Digital Marketing  
Summer Training  
Industrial Training  
College Campus Training

#### CONTACT

Address: G-13, 2nd Floor, Sec-3  
Noida, UP, 201301, India  
Contact No: 0120-4256464, 9990449935  
Contact Us  
Subscribe Us  
Privacy Policy  
Sitemap