

## **DETECT LABELS FROM AN OBJECT STORED IN AWS S3**

## **ABSTRACT**

Amazon Rekognition is based on the same proven, highly scalable, deep learning technology developed by Amazon's computer vision scientists to analyze billions of images and videos daily, and requires no machine learning expertise to use. Amazon Rekognition is a simple and easy to use API that can quickly analyze any image or video file stored in Amazon S3. Amazon Rekognition is always learning from new data, and we are continually adding new labels and facial recognition features to the service. Amazon Rekognition is an image analysis service available in the Amazon AI suite. Through the Amazon Rekognition API, enterprises can enable their applications to detect and analyze scenes, objects, faces and other items within images. Rekognition uses the same machine learning and deep learning technologies as other artificial intelligence (AI) services from Amazon.

# INTRODUCTION

Amazon Rekognition is an image analysis service available in the Amazon AI suite. Through the Amazon Rekognition API, enterprises can enable their applications to detect and analyze scenes, objects, faces and other items within images. Rekognition uses the same machine learning and deep learning technologies as other artificial intelligence (AI) services from Amazon.

The services used are:

- **EC2:-** Amazon Elastic Compute Cloud (Amazon EC2) is a web-based service that allows businesses to run application programs in the Amazon Web Services (AWS) public cloud. Amazon EC2 allows a developer to spin up virtual machines (VM), which provide compute capacity for IT projects and cloud workloads that run with global AWS data centers.
- **Simple Storage Service:-** Amazon Simple Storage Service (Amazon S3) is a scalable, high-speed, low-cost, web-based cloud storage service designed for online backup and archiving of data and application programs. S3 was designed with a minimal feature set and created to make web-scale computing easier for developers.
- **AWS Rekognition:-** Amazon Rekognition makes it easy to add image and video analysis to your applications. You just provide an image or video to the Rekognition API, and the service can identify objects, people, text, scenes, and activities. It can detect any inappropriate content as well. Amazon Rekognition also provides highly accurate facial analysis and facial recognition. You can detect, analyze, and compare faces for a wide variety of use cases, including user verification, cataloging, people counting, and public safety. Amazon Rekognition is based on the same proven, highly scalable, deep learning technology developed by Amazon's computer vision scientists to analyze billions of images and videos daily—and requires no machine learning expertise to use. Amazon Rekognition includes a simple, easy-to-use API that can quickly analyze any image or video file that's stored in Amazon S3.

## STARTING AN INSTANCE

**LOGIN:-**The user must login to the amazon web services using user id and password and if already registered they can sign up directly.



### Root user sign in ⓘ

Email: gtmithunraj99@gmail.com

Password

[Forgot password?](#)

.....

Sign in

[Sign in to a different account](#)

[Create a new AWS account](#)

**CREATE INSTANCE:-** Before you launch and connect to an Amazon EC2 instance, you need to create a key pair, unless you already have one. You can create a key pair using the Amazon EC2 console and then you can launch your EC2 instance.

## Create Instance

---

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Launch Instance



Note: Your instances will launch in the US East (Ohio) region

**CHOOSE AN AMAZON MACHINE IMAGE:-** Amazon EMR uses Amazon Machine Images (AMIs) to initialize the EC2 instances it launches to run a cluster. The AMIs contain the Linux operating system and other software used to run the cluster.

The screenshot shows the AWS console interface for selecting an Amazon Machine Image (AMI). The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and a user profile 'mithunraj99' from 'Ohio'. Below the navigation bar is a progress bar with seven steps: '1. Choose AMI' (active), '2. Choose Instance Type', '3. Configure Instance', '4. Add Storage', '5. Add Tags', '6. Configure Security Group', and '7. Review'. The main heading is 'Step 1: Choose an Amazon Machine Image (AMI)' with a 'Cancel and Exit' link. A sub-heading explains that an AMI is a template containing software configuration. Below this is a 'Quick Start' sidebar with options for 'My AMIs', 'AWS Marketplace', and 'Community AMIs', along with a 'Free tier only' filter. The main content area displays a list of AMIs, with the first three visible: 'Amazon Linux AMI 2017.09.1 (HVM), SSD Volume Type - ami-e81b308d', 'Amazon Linux 2 LTS Candidate AMI 2017.12.0 (HVM), SSD Volume Type - ami-e97c548c', and 'Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-82f4dae7'. Each entry includes a description, root device type, virtualization type, and a 'Select' button. The list is paginated to show '1 to 35 of 35 AMIs'.

AMI Name	AMI ID	Architecture
Amazon Linux AMI 2017.09.1 (HVM), SSD Volume Type	ami-e81b308d	64-bit
Amazon Linux 2 LTS Candidate AMI 2017.12.0 (HVM), SSD Volume Type	ami-e97c548c	64-bit
Ubuntu Server 16.04 LTS (HVM), SSD Volume Type	ami-82f4dae7	64-bit

**CHOOSING AN INSTANCE TYPE:-** General-Purpose. This family includes the M1 and M3 instance types, both of which provide a balance of CPU, memory, and network resources making them a good choice for many applications.

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance types Current generation [Show/Hide Columns](#)

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro <span>Free tier eligible</span>	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Instance Details](#)

**ADDING STORAGE:-** adding storage device to the instance created

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-0bc297f6637ec77c6	8	General Purpose SSD (GP2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)

## CONFIGURE SECURITY GROUP:- Creating the new security group

**Step 6: Configure Security Group**

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group ☐ Select an existing security group

Security group name:

Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

[Add Rule](#)

**Warning**

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[Cancel](#) [Previous](#) [Review and Launch](#)

## Review instance launch:- the instance is review as created

**Step 7: Review Instance Launch**

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

**Improve your instances' security. Your security group, launch-wizard-1, is open to the world.**

Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only. You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

▼ **AMI Details** [Edit AMI](#)

**Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-82f4dae7**

Free tier eligible

Ubuntu Server 16.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

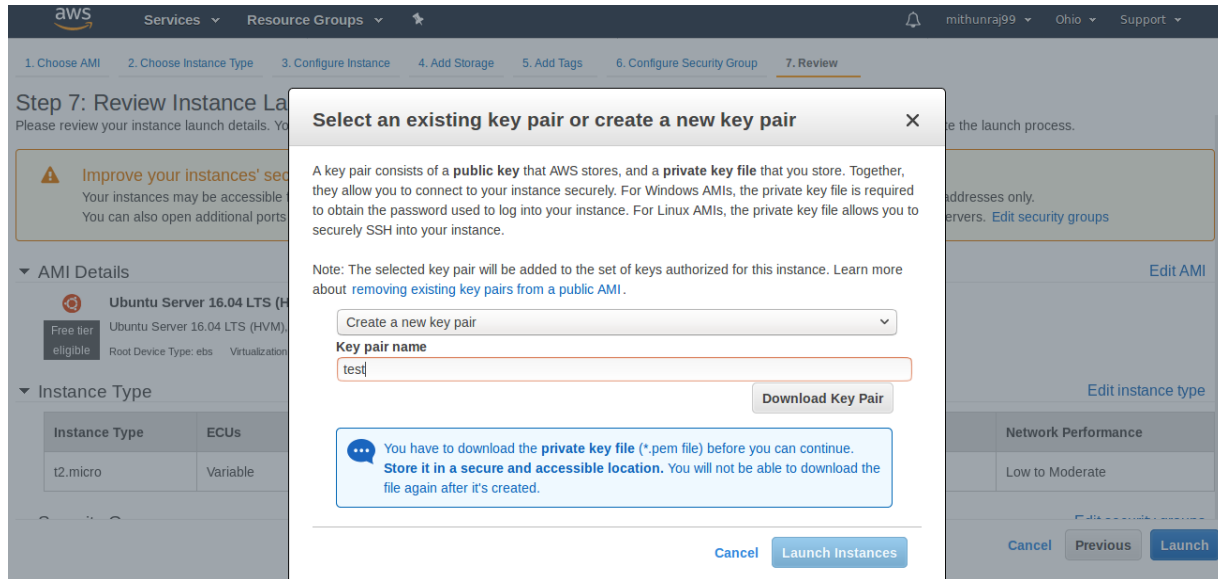
Root Device Type: ebs Virtualization type: hvm

▼ **Instance Type** [Edit instance type](#)

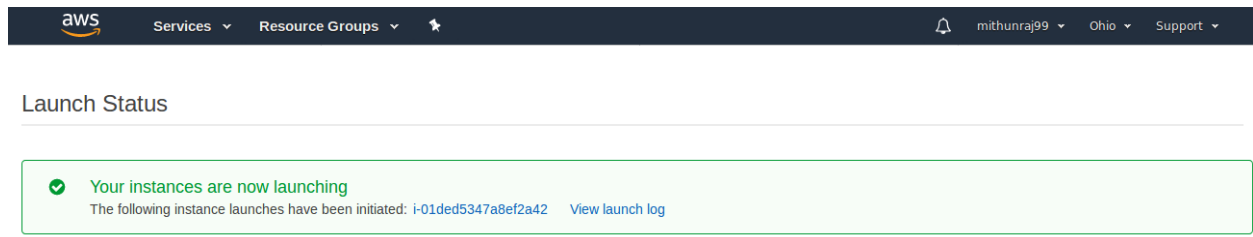
Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

[Cancel](#) [Previous](#) [Launch](#)

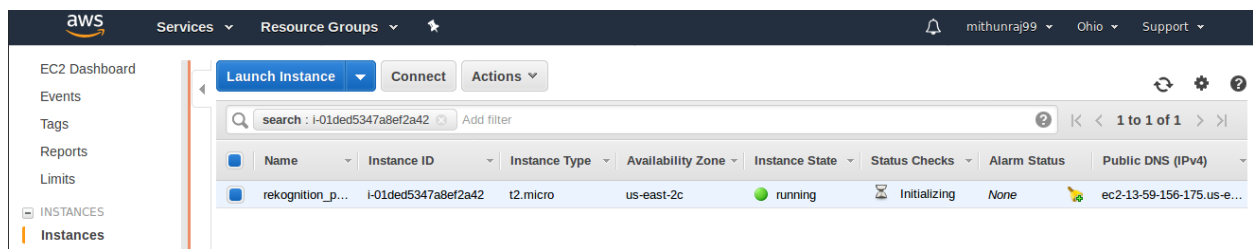
## Key pair is created and launched:



## Launching the status:



## Successfully launched instance and ready to use





## Created instances

The screenshot shows the AWS Management Console interface for the 'Created instances' page. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and user information. The left sidebar lists navigation options: EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES (highlighted), Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, IMAGES, AMIs, Bundle Tasks, ELASTIC BLOCK STORE, Volumes, and Snapshots. The main content area shows a table of instances with columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS (IPv4). One instance is listed: 'rekognition\_p...' with ID 'i-01ded5347a8ef2a42', type 't2.micro', zone 'us-east-2c', state 'running', and public DNS 'ec2-13-59-156-175.us-east-2.compute.amazonaws.com'. Below the table, a detailed view for instance 'i-01ded5347a8ef2a42 (rekognition\_project)' is shown, including tabs for Description, Status Checks, Monitoring, and Tags. The Description tab displays details like Instance ID, Instance state (running), Instance type (t2.micro), Elastic IPs, Availability zone (us-east-2c), Security groups (launch-wizard-1), Scheduled events (No scheduled events), AMI ID (ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20171121.1), Public DNS (IPv4) (ec2-13-59-156-175.us-east-2.compute.amazonaws.com), IPv4 Public IP (13.59.156.175), Private DNS (ip-172-31-46-219.us-east-2.compute.internal), Private IPs (172.31.46.219), VPC ID (vpc-9a4355f3), and Subnet ID (subnet-dc1a7d91).

## Standalone SSH Client Connecting to the created instance:-

**Connect To Your Instance**

I would like to connect with ☒ A standalone SSH client  
☐ A Java SSH Client directly from my browser (Java required)

**To access your instance:**

1. Open an SSH client. (find out how to [connect using PuTTY](#))
2. Locate your private key file (test.pem). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:  

```
chmod 400 test.pem
```
4. Connect to your instance using its Public DNS:  

```
ec2-13-59-156-175.us-east-2.compute.amazonaws.com
```

**Example:**

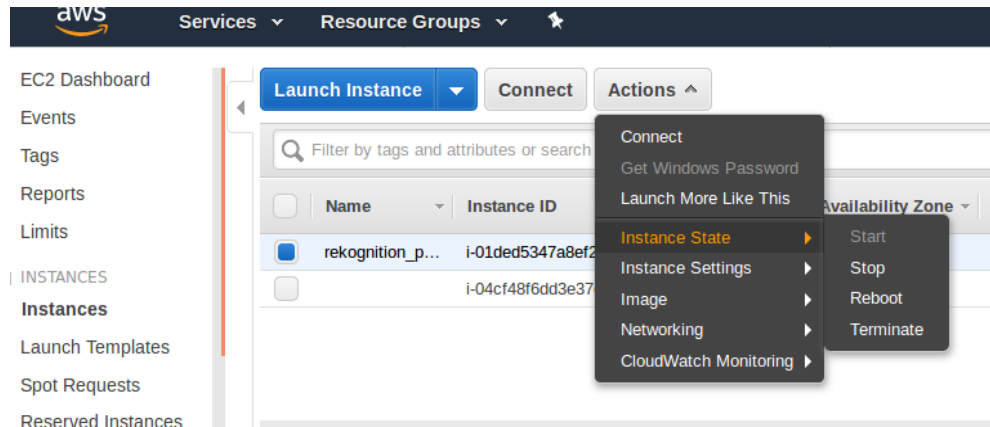
```
ssh -i "test.pem" ubuntu@ec2-13-59-156-175.us-east-2.compute.amazonaws.com
```

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

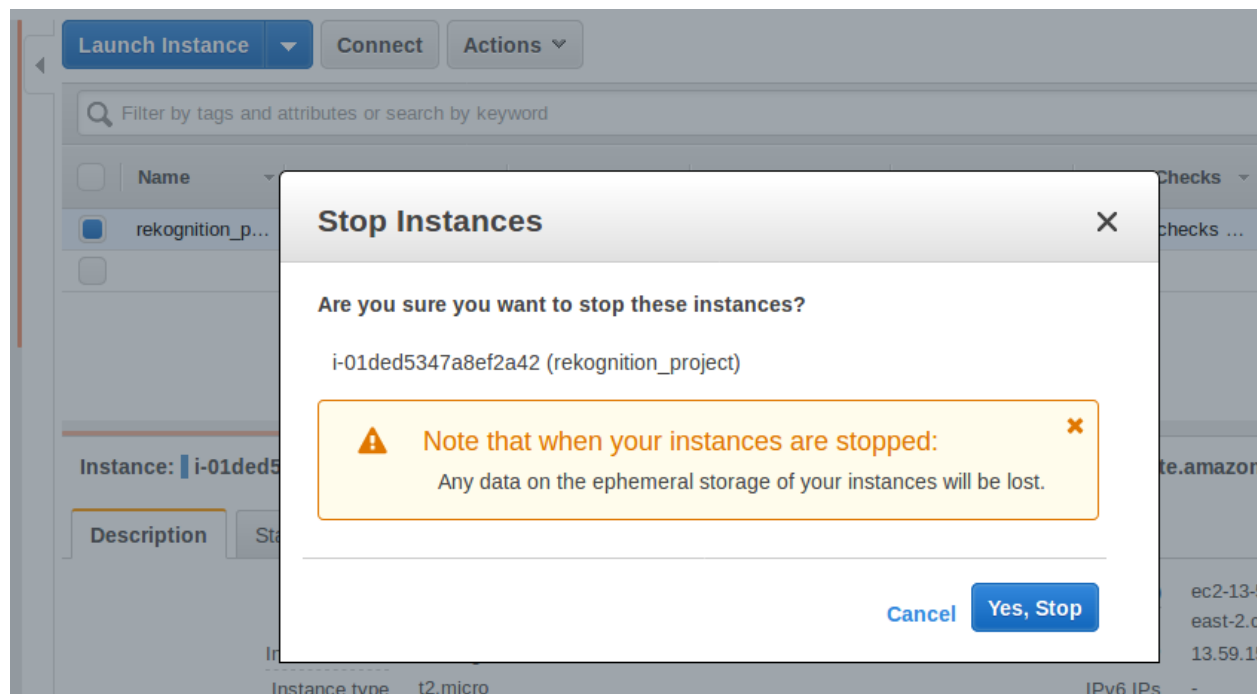
If you need any assistance connecting to your instance, please see our [connection documentation](#).

**Close**



**Properties of instance such as stop ,terminate,create:-**




**How the stop the created instance ?**



## Storing the images in s3 services:-

 Services ▾ Resource Groups ▾ 

 mithunraj99 ▾ Global ▾ Support ▾

Welcome to Amazon S3. Create new buckets or select an existing bucket to view and configure properties. [Documentation](#)


Amazon S3



Discover the new console



Quick tips

 Search for buckets

 Create bucket

Delete bucket

Empty bucket


2 Buckets

0

Public


1 Regions



Bucket name 

Access  

Region 

Date created 

 gtmdemo

Not public \*

US East (N. Virginia)

Dec 31, 2017 1:11:53 AM  
GMT+0530

Amazon S3 > gtmdemo

Overview

Properties

Permissions

Management

 Type a prefix and press Enter to search. Press ESC to clear.



Upload











 Create folder

More ▾

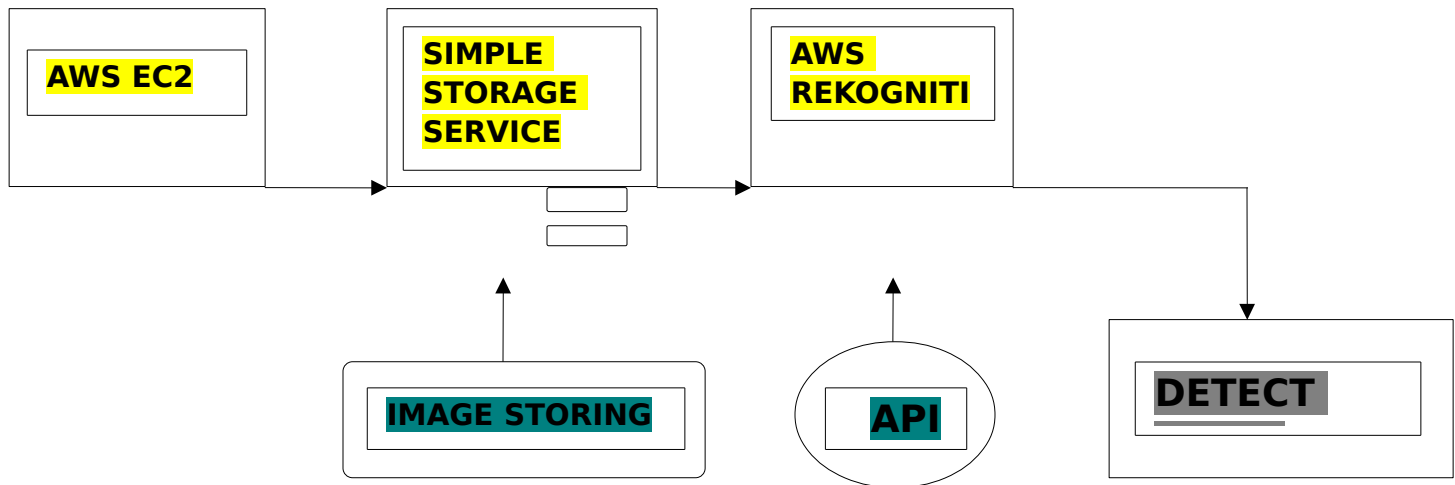
US East (N. Virginia)



Viewing 1 to 13

<input type="checkbox"/> Name 	Last modified 	Size 	Storage class 
<input type="checkbox"/>  God.jpg	Jan 7, 2018 6:26:30 PM GMT+0530	7.7 KB	Standard
<input type="checkbox"/>  bike.jpg	Jan 7, 2018 6:25:47 PM GMT+0530	29.9 KB	Standard
<input type="checkbox"/>  car.jpg	Jan 7, 2018 6:25:49 PM GMT+0530	72.1 KB	Standard
<input type="checkbox"/>  car2.jpg	Jan 7, 2018 6:25:51 PM GMT+0530	71.4 KB	Standard
<input type="checkbox"/>  dogs.jpg	Jan 7, 2018 6:25:53 PM GMT+0530	26.1 KB	Standard
<input type="checkbox"/>  fish.jpg	Jan 7, 2018 6:25:56 PM GMT+0530	247.7 KB	Standard

## Project Working Flow Diagram



## Code execution:-

```
Open ▾ [?]
1 import boto3
2 if __name__ == "__main__":
3     fileName='car.jpg'
4     bucket='gtmdemo'
5     client=boto3.client('rekognition','us-east-1')
6     response = client.detect_labels(Image={'S3Object':{'Bucket':bucket,'Name':fileName}},MinConfidence=40)
7     print('Detected labels for ' + fileName)
8     for label in response['Labels']:
9         print (label['Name'] + ' : ' + str(label['Confidence']))
10
11
```

```
mithun@mithun-HP-Pavilion-15-Notebook-PC: ~
mithun@mithun-HP-Pavilion-15-Notebook-PC:~$ python demo.py
Detected labels for car.jpg
Automobile : 90.1487884521
Car : 90.1487884521
Sedan : 90.1487884521
Transportation : 90.1487884521
Vehicle : 90.1487884521
Coupe : 62.4882583618
Sports Car : 62.4882583618
Suv : 53.4130096436
Van : 52.014881134
Caravan : 50.8963775635
Bumper : 50.819442749
mithun@mithun-HP-Pavilion-15-Notebook-PC:~$
```