
Amazon Rekognition

Amazon Rekognition

API Version 2016-06-27



Amazon Rekognition: Amazon Rekognition

Copyright © 2022 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Welcome	1
Actions	3
CompareFaces	5
Request Syntax	5
Request Parameters	6
Response Syntax	7
Response Elements	8
Errors	9
See Also	10
CopyProjectVersion	11
Request Syntax	11
Request Parameters	11
Response Syntax	13
Response Elements	13
Errors	13
See Also	14
CreateCollection	16
Request Syntax	16
Request Parameters	16
Response Syntax	17
Response Elements	17
Errors	17
See Also	18
CreateDataset	19
Request Syntax	19
Request Parameters	19
Response Syntax	20
Response Elements	20
Errors	20
See Also	21
CreateProject	23
Request Syntax	23
Request Parameters	23
Response Syntax	23
Response Elements	23
Errors	24
See Also	24
CreateProjectVersion	26
Request Syntax	26
Request Parameters	27
Response Syntax	28
Response Elements	29
Errors	29
See Also	30
CreateStreamProcessor	31
Request Syntax	31
Request Parameters	32
Response Syntax	34
Response Elements	34
Errors	35
See Also	36
DeleteCollection	37
Request Syntax	37
Request Parameters	37

Response Syntax	37
Response Elements	37
Errors	37
See Also	38
DeleteDataset	39
Request Syntax	39
Request Parameters	39
Response Elements	39
Errors	39
See Also	40
DeleteFaces	41
Request Syntax	41
Request Parameters	41
Response Syntax	41
Response Elements	41
Errors	42
See Also	42
DeleteProject	44
Request Syntax	44
Request Parameters	44
Response Syntax	44
Response Elements	44
Errors	45
See Also	45
DeleteProjectPolicy	47
Request Syntax	47
Request Parameters	47
Response Elements	48
Errors	48
See Also	48
DeleteProjectVersion	50
Request Syntax	50
Request Parameters	50
Response Syntax	50
Response Elements	50
Errors	51
See Also	51
DeleteStreamProcessor	53
Request Syntax	53
Request Parameters	53
Response Elements	53
Errors	53
See Also	54
DescribeCollection	55
Request Syntax	55
Request Parameters	55
Response Syntax	55
Response Elements	55
Errors	56
See Also	57
DescribeDataset	58
Request Syntax	58
Request Parameters	58
Response Syntax	58
Response Elements	58
Errors	59
See Also	59

DescribeProjects	61
Request Syntax	61
Request Parameters	61
Response Syntax	62
Response Elements	62
Errors	62
See Also	63
DescribeProjectVersions	64
Request Syntax	64
Request Parameters	64
Response Syntax	65
Response Elements	67
Errors	67
See Also	68
DescribeStreamProcessor	69
Request Syntax	69
Request Parameters	69
Response Syntax	69
Response Elements	70
Errors	72
See Also	73
DetectCustomLabels	74
Request Syntax	74
Request Parameters	74
Response Syntax	76
Response Elements	76
Errors	76
See Also	77
DetectFaces	79
Request Syntax	79
Request Parameters	79
Response Syntax	80
Response Elements	81
Errors	81
See Also	82
DetectLabels	84
Request Syntax	85
Request Parameters	86
Response Syntax	87
Response Elements	88
Errors	89
See Also	90
DetectModerationLabels	91
Request Syntax	91
Request Parameters	91
Response Syntax	92
Response Elements	92
Errors	93
See Also	93
DetectProtectiveEquipment	95
Request Syntax	95
Request Parameters	95
Response Syntax	96
Response Elements	97
Errors	97
See Also	98
DetectText	99

Request Syntax	99
Request Parameters	100
Response Syntax	100
Response Elements	101
Errors	101
See Also	102
DistributeDatasetEntries	103
Request Syntax	103
Request Parameters	103
Response Elements	103
Errors	103
See Also	104
GetCelebrityInfo	105
Request Syntax	105
Request Parameters	105
Response Syntax	105
Response Elements	105
Errors	106
See Also	106
GetCelebrityRecognition	108
Request Syntax	108
Request Parameters	108
Response Syntax	109
Response Elements	111
Errors	112
See Also	112
GetContentModeration	114
Request Syntax	114
Request Parameters	114
Response Syntax	115
Response Elements	116
Errors	116
See Also	117
GetFaceDetection	118
Request Syntax	118
Request Parameters	118
Response Syntax	119
Response Elements	120
Errors	121
See Also	122
GetFaceSearch	123
Request Syntax	123
Request Parameters	123
Response Syntax	124
Response Elements	126
Errors	127
See Also	127
GetLabelDetection	129
Request Syntax	129
Request Parameters	129
Response Syntax	130
Response Elements	131
Errors	132
See Also	133
GetPersonTracking	134
Request Syntax	134
Request Parameters	134

Response Syntax	135
Response Elements	137
Errors	137
See Also	138
GetSegmentDetection	139
Request Syntax	139
Request Parameters	139
Response Syntax	140
Response Elements	141
Errors	142
See Also	142
GetTextDetection	144
Request Syntax	144
Request Parameters	144
Response Syntax	145
Response Elements	146
Errors	146
See Also	147
IndexFaces	148
Request Syntax	149
Request Parameters	149
Response Syntax	151
Response Elements	153
Errors	154
See Also	155
ListCollections	157
Request Syntax	157
Request Parameters	157
Response Syntax	157
Response Elements	157
Errors	158
See Also	159
ListDatasetEntries	160
Request Syntax	160
Request Parameters	160
Response Syntax	162
Response Elements	162
Errors	162
See Also	163
ListDatasetLabels	164
Request Syntax	164
Request Parameters	164
Response Syntax	165
Response Elements	165
Errors	165
See Also	166
ListFaces	167
Request Syntax	167
Request Parameters	167
Response Syntax	168
Response Elements	168
Errors	168
See Also	169
ListProjectPolicies	170
Request Syntax	170
Request Parameters	170
Response Syntax	171

Response Elements	171
Errors	171
See Also	172
ListStreamProcessors	173
Request Syntax	173
Request Parameters	173
Response Syntax	173
Response Elements	173
Errors	174
See Also	174
ListTagsForResource	176
Request Syntax	176
Request Parameters	176
Response Syntax	176
Response Elements	176
Errors	177
See Also	177
PutProjectPolicy	179
Request Syntax	179
Request Parameters	179
Response Syntax	180
Response Elements	180
Errors	180
See Also	182
RecognizeCelebrities	183
Request Syntax	183
Request Parameters	183
Response Syntax	184
Response Elements	185
Errors	186
See Also	187
SearchFaces	188
Request Syntax	188
Request Parameters	188
Response Syntax	189
Response Elements	189
Errors	190
See Also	190
SearchFacesByImage	192
Request Syntax	192
Request Parameters	193
Response Syntax	194
Response Elements	194
Errors	195
See Also	196
StartCelebrityRecognition	197
Request Syntax	197
Request Parameters	197
Response Syntax	198
Response Elements	198
Errors	198
See Also	199
StartContentModeration	201
Request Syntax	201
Request Parameters	201
Response Syntax	202
Response Elements	202

Errors	203
See Also	204
StartFaceDetection	205
Request Syntax	205
Request Parameters	205
Response Syntax	206
Response Elements	206
Errors	207
See Also	208
StartFaceSearch	209
Request Syntax	209
Request Parameters	209
Response Syntax	210
Response Elements	210
Errors	211
See Also	212
StartLabelDetection	213
Request Syntax	213
Request Parameters	213
Response Syntax	214
Response Elements	214
Errors	215
See Also	216
StartPersonTracking	217
Request Syntax	217
Request Parameters	217
Response Syntax	218
Response Elements	218
Errors	218
See Also	219
StartProjectVersion	221
Request Syntax	221
Request Parameters	221
Response Syntax	222
Response Elements	222
Errors	222
See Also	223
StartSegmentDetection	224
Request Syntax	224
Request Parameters	224
Response Syntax	226
Response Elements	226
Errors	226
See Also	227
StartStreamProcessor	228
Request Syntax	228
Request Parameters	228
Response Syntax	229
Response Elements	229
Errors	229
See Also	230
StartTextDetection	231
Request Syntax	231
Request Parameters	231
Response Syntax	232
Response Elements	233
Errors	233

See Also	234
StopProjectVersion	235
Request Syntax	235
Request Parameters	235
Response Syntax	235
Response Elements	235
Errors	236
See Also	236
StopStreamProcessor	238
Request Syntax	238
Request Parameters	238
Response Elements	238
Errors	238
See Also	239
TagResource	240
Request Syntax	240
Request Parameters	240
Response Elements	240
Errors	241
See Also	241
UntagResource	243
Request Syntax	243
Request Parameters	243
Response Elements	243
Errors	243
See Also	244
UpdateDatasetEntries	245
Request Syntax	245
Request Parameters	245
Response Elements	246
Errors	246
See Also	247
UpdateStreamProcessor	248
Request Syntax	248
Request Parameters	248
Response Elements	249
Errors	249
See Also	250
Data Types	251
AgeRange	254
Contents	254
See Also	254
Asset	255
Contents	255
See Also	255
AudioMetadata	256
Contents	256
See Also	256
Beard	257
Contents	257
See Also	257
BlackFrame	258
Contents	258
See Also	258
BoundingBox	259
Contents	259
See Also	259

Celebrity	261
Contents	261
See Also	262
CelebrityDetail	263
Contents	263
See Also	264
CelebrityRecognition	265
Contents	265
See Also	265
ComparedFace	266
Contents	266
See Also	267
ComparedSourceImageFace	268
Contents	268
See Also	268
CompareFacesMatch	269
Contents	269
See Also	269
ConnectedHomeSettings	270
Contents	270
See Also	270
ConnectedHomeSettingsForUpdate	271
Contents	271
See Also	271
ContentModerationDetection	272
Contents	272
See Also	272
CoversBodyPart	273
Contents	273
See Also	273
CustomLabel	274
Contents	274
See Also	274
DatasetChanges	275
Contents	275
See Also	275
DatasetDescription	276
Contents	276
See Also	277
DatasetLabelDescription	278
Contents	278
See Also	278
DatasetLabelStats	279
Contents	279
See Also	279
DatasetMetadata	280
Contents	280
See Also	281
DatasetSource	282
Contents	282
See Also	282
DatasetStats	283
Contents	283
See Also	283
DetectionFilter	284
Contents	284
See Also	284

DetectLabelsImageBackground	285
Contents	285
See Also	285
DetectLabelsImageForeground	286
Contents	286
See Also	286
DetectLabelsImageProperties	287
Contents	287
See Also	287
DetectLabelsImagePropertiesSettings	288
Contents	288
See Also	288
DetectLabelsImageQuality	289
Contents	289
See Also	289
DetectLabelsSettings	290
Contents	290
See Also	290
DetectTextFilters	291
Contents	291
See Also	291
DistributeDataset	292
Contents	292
See Also	292
DominantColor	293
Contents	293
See Also	294
Emotion	295
Contents	295
See Also	295
EquipmentDetection	296
Contents	296
See Also	296
EvaluationResult	297
Contents	297
See Also	297
Eyeglasses	298
Contents	298
See Also	298
EyeOpen	299
Contents	299
See Also	299
Face	300
Contents	300
See Also	301
FaceDetail	302
Contents	302
See Also	304
FaceDetection	305
Contents	305
See Also	305
FaceMatch	306
Contents	306
See Also	306
FaceRecord	307
Contents	307
See Also	307

FaceSearchSettings	308
Contents	308
See Also	308
Gender	309
Contents	309
See Also	309
GeneralLabelsSettings	310
Contents	310
See Also	311
Geometry	312
Contents	312
See Also	312
GroundTruthManifest	313
Contents	313
See Also	313
HumanLoopActivationOutput	314
Contents	314
See Also	314
HumanLoopConfig	315
Contents	315
See Also	315
HumanLoopDataAttributes	316
Contents	316
See Also	316
Image	317
Contents	317
See Also	317
ImageQuality	318
Contents	318
See Also	318
Instance	319
Contents	319
See Also	319
KinesisDataStream	320
Contents	320
See Also	320
KinesisVideoStream	321
Contents	321
See Also	321
KinesisVideoStreamStartSelector	322
Contents	322
See Also	322
KnownGender	323
Contents	323
See Also	323
Label	324
Contents	324
See Also	324
LabelAlias	326
Contents	326
See Also	326
LabelCategory	327
Contents	327
See Also	327
LabelDetection	328
Contents	328
See Also	328

Landmark	329
Contents	329
See Also	329
ModerationLabel	330
Contents	330
See Also	330
MouthOpen	331
Contents	331
See Also	331
Mustache	332
Contents	332
See Also	332
NotificationChannel	333
Contents	333
See Also	333
OutputConfig	334
Contents	334
See Also	334
Parent	335
Contents	335
See Also	335
PersonDetail	336
Contents	336
See Also	336
PersonDetection	337
Contents	337
See Also	337
PersonMatch	338
Contents	338
See Also	338
Point	339
Contents	339
See Also	339
Pose	340
Contents	340
See Also	340
ProjectDescription	341
Contents	341
See Also	341
ProjectPolicy	342
Contents	342
See Also	343
ProjectVersionDescription	344
Contents	344
See Also	346
ProtectiveEquipmentBodyPart	347
Contents	347
See Also	347
ProtectiveEquipmentPerson	348
Contents	348
See Also	348
ProtectiveEquipmentSummarizationAttributes	349
Contents	349
See Also	349
ProtectiveEquipmentSummary	350
Contents	350
See Also	350

RegionOfInterest	352
Contents	352
See Also	352
S3Destination	353
Contents	353
See Also	353
S3Object	354
Contents	354
See Also	354
SegmentDetection	355
Contents	355
See Also	356
SegmentTypeInfo	358
Contents	358
See Also	358
ShotSegment	359
Contents	359
See Also	359
Smile	360
Contents	360
See Also	360
StartSegmentDetectionFilters	361
Contents	361
See Also	361
StartShotDetectionFilter	362
Contents	362
See Also	362
StartTechnicalCueDetectionFilter	363
Contents	363
See Also	363
StartTextDetectionFilters	364
Contents	364
See Also	364
StreamProcessingStartSelector	365
Contents	365
See Also	365
StreamProcessingStopSelector	366
Contents	366
See Also	366
StreamProcessor	367
Contents	367
See Also	367
StreamProcessorDataSharingPreference	368
Contents	368
See Also	368
StreamProcessorInput	369
Contents	369
See Also	369
StreamProcessorNotificationChannel	370
Contents	370
See Also	370
StreamProcessorOutput	371
Contents	371
See Also	371
StreamProcessorSettings	372
Contents	372
See Also	372

StreamProcessorSettingsForUpdate	373
Contents	373
See Also	373
Summary	374
Contents	374
See Also	374
Sunglasses	375
Contents	375
See Also	375
TechnicalCueSegment	376
Contents	376
See Also	376
TestingData	377
Contents	377
See Also	377
TestingDataResult	378
Contents	378
See Also	378
TextDetection	379
Contents	379
See Also	380
TextDetectionResult	381
Contents	381
See Also	381
TrainingData	382
Contents	382
See Also	382
TrainingDataResult	383
Contents	383
See Also	383
UnindexedFace	384
Contents	384
See Also	384
ValidationData	385
Contents	385
See Also	385
Video	386
Contents	386
See Also	386
VideoMetadata	387
Contents	387
See Also	388
Common Parameters	389
Common Errors	391

Welcome

This is the API Reference for [Amazon Rekognition Image](#), [Amazon Rekognition Custom Labels](#), [Amazon Rekognition Stored Video](#), [Amazon Rekognition Streaming Video](#). It provides descriptions of actions, data types, common parameters, and common errors.

Amazon Rekognition Image

- [CompareFaces](#)
- [CreateCollection](#)
- [DeleteCollection](#)
- [DeleteFaces](#)
- [DescribeCollection](#)
- [DetectFaces](#)
- [DetectLabels](#)
- [DetectModerationLabels](#)
- [DetectProtectiveEquipment](#)
- [DetectText](#)
- [GetCelebrityInfo](#)
- [IndexFaces](#)
- [ListCollections](#)
- [ListFaces](#)
- [RecognizeCelebrities](#)
- [SearchFaces](#)
- [SearchFacesByImage](#)

Amazon Rekognition Custom Labels

- [CopyProjectVersion](#)
- [CreateDataset](#)
- [CreateProject](#)
- [CreateProjectVersion](#)
- [DeleteDataset](#)
- [DeleteProject](#)
- [DeleteProjectPolicy](#)
- [DeleteProjectVersion](#)
- [DescribeDataset](#)
- [DescribeProjects](#)
- [DescribeProjectVersions](#)
- [DetectCustomLabels](#)
- [DistributeDatasetEntries](#)
- [ListDatasetEntries](#)
- [ListDatasetLabels](#)
- [ListProjectPolicies](#)
- [PutProjectPolicy](#)

- [StartProjectVersion](#)
- [StopProjectVersion](#)
- [UpdateDatasetEntries](#)

Amazon Rekognition Video Stored Video

- [GetCelebrityRecognition](#)
- [GetContentModeration](#)
- [GetFaceDetection](#)
- [GetFaceSearch](#)
- [GetLabelDetection](#)
- [GetPersonTracking](#)
- [GetSegmentDetection](#)
- [GetTextDetection](#)
- [StartCelebrityRecognition](#)
- [StartContentModeration](#)
- [StartFaceDetection](#)
- [StartFaceSearch](#)
- [StartLabelDetection](#)
- [StartPersonTracking](#)
- [StartSegmentDetection](#)
- [StartTextDetection](#)

Amazon Rekognition Video Streaming Video

- [CreateStreamProcessor](#)
- [DeleteStreamProcessor](#)
- [DescribeStreamProcessor](#)
- [ListStreamProcessors](#)
- [StartStreamProcessor](#)
- [StopStreamProcessor](#)
- [UpdateStreamProcessor](#)

This document was last published on November 23, 2022.

Actions

The following actions are supported:

- [CompareFaces \(p. 5\)](#)
- [CopyProjectVersion \(p. 11\)](#)
- [CreateCollection \(p. 16\)](#)
- [CreateDataset \(p. 19\)](#)
- [CreateProject \(p. 23\)](#)
- [CreateProjectVersion \(p. 26\)](#)
- [CreateStreamProcessor \(p. 31\)](#)
- [DeleteCollection \(p. 37\)](#)
- [DeleteDataset \(p. 39\)](#)
- [DeleteFaces \(p. 41\)](#)
- [DeleteProject \(p. 44\)](#)
- [DeleteProjectPolicy \(p. 47\)](#)
- [DeleteProjectVersion \(p. 50\)](#)
- [DeleteStreamProcessor \(p. 53\)](#)
- [DescribeCollection \(p. 55\)](#)
- [DescribeDataset \(p. 58\)](#)
- [DescribeProjects \(p. 61\)](#)
- [DescribeProjectVersions \(p. 64\)](#)
- [DescribeStreamProcessor \(p. 69\)](#)
- [DetectCustomLabel \(p. 74\)](#)
- [DetectFaces \(p. 79\)](#)
- [DetectLabels \(p. 84\)](#)
- [DetectModerationLabels \(p. 91\)](#)
- [DetectProtectiveEquipment \(p. 95\)](#)
- [DetectText \(p. 99\)](#)
- [DistributeDatasetEntries \(p. 103\)](#)
- [GetCelebrityInfo \(p. 105\)](#)
- [GetCelebrityRecognition \(p. 108\)](#)
- [GetContentModeration \(p. 114\)](#)
- [GetFaceDetection \(p. 118\)](#)
- [GetFaceSearch \(p. 123\)](#)
- [GetLabelDetection \(p. 129\)](#)
- [GetPersonTracking \(p. 134\)](#)
- [GetSegmentDetection \(p. 139\)](#)
- [GetTextDetection \(p. 144\)](#)
- [IndexFaces \(p. 148\)](#)
- [ListCollections \(p. 157\)](#)
- [ListDatasetEntries \(p. 160\)](#)
- [ListDatasetLabels \(p. 164\)](#)
- [ListFaces \(p. 167\)](#)

- [ListProjectPolicies \(p. 170\)](#)
- [ListStreamProcessors \(p. 173\)](#)
- [ListTagsForResource \(p. 176\)](#)
- [PutProjectPolicy \(p. 179\)](#)
- [RecognizeCelebrities \(p. 183\)](#)
- [SearchFaces \(p. 188\)](#)
- [SearchFacesByImage \(p. 192\)](#)
- [StartCelebrityRecognition \(p. 197\)](#)
- [StartContentModeration \(p. 201\)](#)
- [StartFaceDetection \(p. 205\)](#)
- [StartFaceSearch \(p. 209\)](#)
- [StartLabelDetection \(p. 213\)](#)
- [StartPersonTracking \(p. 217\)](#)
- [StartProjectVersion \(p. 221\)](#)
- [StartSegmentDetection \(p. 224\)](#)
- [StartStreamProcessor \(p. 228\)](#)
- [StartTextDetection \(p. 231\)](#)
- [StopProjectVersion \(p. 235\)](#)
- [StopStreamProcessor \(p. 238\)](#)
- [TagResource \(p. 240\)](#)
- [UntagResource \(p. 243\)](#)
- [UpdateDatasetEntries \(p. 245\)](#)
- [UpdateStreamProcessor \(p. 248\)](#)

CompareFaces

Compares a face in the *source* input image with each of the 100 largest faces detected in the *target* input image.

If the source image contains multiple faces, the service detects the largest face and compares it with each face detected in the target image.

Note

CompareFaces uses machine learning algorithms, which are probabilistic. A false negative is an incorrect prediction that a face in the target image has a low similarity confidence score when compared to the face in the source image. To reduce the probability of false negatives, we recommend that you compare the target image against multiple source images. If you plan to use CompareFaces to make a decision that impacts an individual's rights, privacy, or access to services, we recommend that you pass the result to a human for review and further validation before taking action.

You pass the input and target images either as base64-encoded image bytes or as references to images in an Amazon S3 bucket. If you use the AWS CLI to call Amazon Rekognition operations, passing image bytes isn't supported. The image must be formatted as a PNG or JPEG file.

In response, the operation returns an array of face matches ordered by similarity score in descending order. For each face match, the response provides a bounding box of the face, facial landmarks, pose details (pitch, roll, and yaw), quality (brightness and sharpness), and confidence value (indicating the level of confidence that the bounding box contains a face). The response also provides a similarity score, which indicates how closely the faces match.

Note

By default, only faces with a similarity score of greater than or equal to 80% are returned in the response. You can change this value by specifying the `SimilarityThreshold` parameter.

CompareFaces also returns an array of faces that don't match the source image. For each face, it returns a bounding box, confidence value, landmarks, pose details, and quality. The response also returns information about the face in the source image, including the bounding box of the face and confidence value.

The `QualityFilter` input parameter allows you to filter out detected faces that don't meet a required quality bar. The quality bar is based on a variety of common use cases. Use `QualityFilter` to set the quality bar by specifying `LOW`, `MEDIUM`, or `HIGH`. If you do not want to filter detected faces, specify `NONE`. The default value is `NONE`.

If the image doesn't contain Exif metadata, CompareFaces returns orientation information for the source and target images. Use these values to display the images with the correct image orientation.

If no faces are detected in the source or target images, CompareFaces returns an `InvalidParameterException` error.

Note

This is a stateless API operation. That is, data returned by this operation doesn't persist.

For an example, see [Comparing faces in images](#).

This operation requires permissions to perform the `rekognition:CompareFaces` action.

Request Syntax

```
{  
  "QualityFilter": "string",  
  "SimilarityThreshold": number,  
}
```

```
"SourceImage": {  
    "Bytes": blob,  
    "S3Object": {  
        "Bucket": "string",  
        "Name": "string",  
        "Version": "string"  
    }  
},  
"TargetImage": {  
    "Bytes": blob,  
    "S3Object": {  
        "Bucket": "string",  
        "Name": "string",  
        "Version": "string"  
    }  
}  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[QualityFilter \(p. 5\)](#)

A filter that specifies a quality bar for how much filtering is done to identify faces. Filtered faces aren't compared. If you specify AUTO, Amazon Rekognition chooses the quality bar. If you specify LOW, MEDIUM, or HIGH, filtering removes all faces that don't meet the chosen quality bar. The quality bar is based on a variety of common use cases. Low-quality detections can occur for a number of reasons. Some examples are an object that's misidentified as a face, a face that's too blurry, or a face with a pose that's too extreme to use. If you specify NONE, no filtering is performed. The default value is NONE.

To use quality filtering, the collection you are using must be associated with version 3 of the face model or higher.

Type: String

Valid Values: NONE | AUTO | LOW | MEDIUM | HIGH

Required: No

[SimilarityThreshold \(p. 5\)](#)

The minimum level of confidence in the face matches that a match must meet to be included in the FaceMatches array.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

[SourceImage \(p. 5\)](#)

The input image as base64-encoded bytes or an S3 object. If you use the AWS CLI to call Amazon Rekognition operations, passing base64-encoded image bytes is not supported.

If you are using an AWS SDK to call Amazon Rekognition, you might not need to base64-encode image bytes passed using the Bytes field. For more information, see [Image specifications](#).

Type: [Image \(p. 317\)](#) object

Required: Yes

[TargetImage \(p. 5\)](#)

The target image as base64-encoded bytes or an S3 object. If you use the AWS CLI to call Amazon Rekognition operations, passing base64-encoded image bytes is not supported.

If you are using an AWS SDK to call Amazon Rekognition, you might not need to base64-encode image bytes passed using the Bytes field. For more information, see [Image specifications](#).

Type: [Image \(p. 317\)](#) object

Required: Yes

Response Syntax

```
{  
  "FaceMatches": [  
    {  
      "Face": {  
        "BoundingBox": {  
          "Height": number,  
          "Left": number,  
          "Top": number,  
          "Width": number  
        },  
        "Confidence": number,  
        "Emotions": [  
          {  
            "Confidence": number,  
            "Type": "string"  
          }  
        ],  
        "Landmarks": [  
          {  
            "Type": "string",  
            "X": number,  
            "Y": number  
          }  
        ],  
        "Pose": {  
          "Pitch": number,  
          "Roll": number,  
          "Yaw": number  
        },  
        "Quality": {  
          "Brightness": number,  
          "Sharpness": number  
        },  
        "Smile": {  
          "Confidence": number,  
          "Value": boolean  
        }  
      },  
      "Similarity": number  
    }  
  ],  
  "SourceImageFace": {  
    "BoundingBox": {  
      "Height": number,  
      "Left": number,  
      "Top": number,  
      "Width": number  
    }  
  }  
}
```

```

        "Top": number,
        "Width": number
    },
    "Confidence": number
},
"SourceImageOrientationCorrection": "string",
"TargetImageOrientationCorrection": "string",
"UnmatchedFaces": [
    {
        "BoundingBox": {
            "Height": number,
            "Left": number,
            "Top": number,
            "Width": number
        },
        "Confidence": number,
        "Emotions": [
            {
                "Confidence": number,
                "Type": "string"
            }
        ],
        "Landmarks": [
            {
                "Type": "string",
                "X": number,
                "Y": number
            }
        ],
        "Pose": {
            "Pitch": number,
            "Roll": number,
            "Yaw": number
        },
        "Quality": {
            "Brightness": number,
            "Sharpness": number
        },
        "Smile": {
            "Confidence": number,
            "Value": boolean
        }
    }
]
}

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

FaceMatches (p. 7)

An array of faces in the target image that match the source image face. Each `CompareFacesMatch` object provides the bounding box, the confidence level that the bounding box contains a face, and the similarity score for the face in the bounding box and the face in the source image.

Type: Array of [CompareFacesMatch \(p. 269\)](#) objects

SourceImageFace (p. 7)

The face in the source image that was used for comparison.

Type: [ComparedSourceImageFace \(p. 268\)](#) object

[SourceImageOrientationCorrection \(p. 7\)](#)

The value of `SourceImageOrientationCorrection` is always null.

If the input image is in .jpeg format, it might contain exchangeable image file format (Exif) metadata that includes the image's orientation. Amazon Rekognition uses this orientation information to perform image correction. The bounding box coordinates are translated to represent object locations after the orientation information in the Exif metadata is used to correct the image orientation. Images in .png format don't contain Exif metadata.

Amazon Rekognition doesn't perform image correction for images in .png format and .jpeg images without orientation information in the image Exif metadata. The bounding box coordinates aren't translated and represent the object locations before the image is rotated.

Type: String

Valid Values: ROTATE_0 | ROTATE_90 | ROTATE_180 | ROTATE_270

[TargetImageOrientationCorrection \(p. 7\)](#)

The value of `TargetImageOrientationCorrection` is always null.

If the input image is in .jpeg format, it might contain exchangeable image file format (Exif) metadata that includes the image's orientation. Amazon Rekognition uses this orientation information to perform image correction. The bounding box coordinates are translated to represent object locations after the orientation information in the Exif metadata is used to correct the image orientation. Images in .png format don't contain Exif metadata.

Amazon Rekognition doesn't perform image correction for images in .png format and .jpeg images without orientation information in the image Exif metadata. The bounding box coordinates aren't translated and represent the object locations before the image is rotated.

Type: String

Valid Values: ROTATE_0 | ROTATE_90 | ROTATE_180 | ROTATE_270

[UnmatchedFaces \(p. 7\)](#)

An array of faces in the target image that did not match the source image face.

Type: Array of [ComparedFace \(p. 266\)](#) objects

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

ImageTooLargeException

The input image size exceeds the allowed limit. If you are calling [DetectProtectiveEquipment \(p. 95\)](#), the image size or resolution exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidImageFormatException

The provided image format is not supported.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

CopyProjectVersion

Copies a version of an Amazon Rekognition Custom Labels model from a source project to a destination project. The source and destination projects can be in different AWS accounts but must be in the same AWS Region. You can't copy a model to another AWS service.

To copy a model version to a different AWS account, you need to create a resource-based policy known as a *project policy*. You attach the project policy to the source project by calling [PutProjectPolicy \(p. 179\)](#). The project policy gives permission to copy the model version from a trusting AWS account to a trusted account.

For more information about creating and attaching a project policy document, see [Attaching a project policy \(SDK\)](#).

If you are copying a model version to a project in the same AWS account, you don't need to create a project policy.

Note

To copy a model, the destination project, source project, and source model version must already exist.

Copying a model version takes a while to complete. To get the current status, call [DescribeProjectVersions \(p. 64\)](#) and check the value of Status in the [ProjectVersionDescription \(p. 344\)](#) object. The copy operation has finished when the value of Status is COPYING_COMPLETED.

Request Syntax

```
{  
    "DestinationProjectArn": "string",  
    "KmsKeyId": "string",  
    "OutputConfig": {  
        "S3Bucket": "string",  
        "S3KeyPrefix": "string"  
    },  
    "SourceProjectArn": "string",  
    "SourceProjectVersionArn": "string",  
    "Tags": {  
        "string" : "string"  
    },  
    "VersionName": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[DestinationProjectArn \(p. 11\)](#)

The ARN of the project in the trusted AWS account that you want to copy the model version to.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-zA-Z0-9_.-]{1,255}:\/\/[0-9]+\$)

Required: Yes

[KmsKeyId \(p. 11\)](#)

The identifier for your AWS Key Management Service key (AWS KMS key). You can supply the Amazon Resource Name (ARN) of your KMS key, the ID of your KMS key, an alias for your KMS key, or an alias ARN. The key is used to encrypt training results and manifest files written to the output Amazon S3 bucket (OutputConfig).

If you choose to use your own KMS key, you need the following permissions on the KMS key.

- kms:CreateGrant
- kms:DescribeKey
- kms:GenerateDataKey
- kms:Decrypt

If you don't specify a value for KmsKeyId, images copied into the service are encrypted using a key that AWS owns and manages.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: ^[A-Za-z0-9][A-Za-z0-9:_+=,@.-]{0,2048}\$

Required: No

[OutputConfig \(p. 11\)](#)

The S3 bucket and folder location where the training output for the source model version is placed.

Type: [OutputConfig \(p. 334\)](#) object

Required: Yes

[SourceProjectArn \(p. 11\)](#)

The ARN of the source project in the trusting AWS account.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-zA-Z0-9_.-]{1,255}:\/\/[0-9]+\$)

Required: Yes

[SourceProjectVersionArn \(p. 11\)](#)

The ARN of the model version in the source project that you want to copy to a destination project.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-zA-Z0-9_.-]{1,255}:\/\/version\/[a-zA-Z0-9_.-]{1,255}\/[0-9]+\$)

Required: Yes

Tags (p. 11)

The key-value tags to assign to the model version.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 200 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `^(?!aws:)[\p{L}\p{Z}\p{N}_.:/=+\-@]*$`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

Required: No

VersionName (p. 11)

A name for the version of the model that's copied to the destination project.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_.\-\u00b7]+`

Required: Yes

Response Syntax

```
{  
  "ProjectVersionArn": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ProjectVersionArn (p. 13)

The ARN of the copied model version in the destination project.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `(^arn:[a-zA-Z0-9\.\-\u00b7]+:rekognition:[a-zA-Z0-9\.\-\u00b7]+\d{12}:project\[/[a-zA-Z0-9\.\-\u00b7]+\{1,255\}\version\[/[a-zA-Z0-9\.\-\u00b7]+\{1,255\}\[/[0-9]+\$)`

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a LimitExceededException exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ServiceQuotaExceededException

The size of the resource exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

CreateCollection

Creates a collection in an AWS Region. You can add faces to the collection using the [IndexFaces \(p. 148\)](#) operation.

For example, you might create collections, one for each of your application users. A user can then index faces using the `IndexFaces` operation and persist results in a specific collection. Then, a user can search the collection for faces in the user-specific container.

When you create a collection, it is associated with the latest version of the face model version.

Note

Collection names are case-sensitive.

This operation requires permissions to perform the `rekognition:CreateCollection` action. If you want to tag your collection, you also require permission to perform the `rekognition:TagResource` operation.

Request Syntax

```
{  
  "CollectionId": "string",  
  "Tags": {  
    "string" : "string"  
  }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

CollectionId (p. 16)

ID for the collection that you are creating.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

Tags (p. 16)

A set of tags (key-value pairs) that you want to attach to the collection.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 200 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: ^(?!aws:)[\p{L}\p{Z}]\p{N}_.:/+=\-\@]*\$

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `^([\p{L}\p{Z}\p{N}_.:=/=+\-@\p{N}]*)$`

Required: No

Response Syntax

```
{  
  "CollectionArn": "string",  
  "FaceModelVersion": "string",  
  "StatusCode": number  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

CollectionArn (p. 17)

Amazon Resource Name (ARN) of the collection. You can use this to manage permissions on your resources.

Type: String

FaceModelVersion (p. 17)

Version number of the face detection model associated with the collection you are creating.

Type: String

StatusCode (p. 17)

HTTP status code indicating the result of the operation.

Type: Integer

Valid Range: Minimum value of 0.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceAlreadyExistsException

A resource with the specified ID already exists.

HTTP Status Code: 400

ServiceQuotaExceededException

The size of the resource exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

CreateDataset

Creates a new Amazon Rekognition Custom Labels dataset. You can create a dataset by using an Amazon Sagemaker format manifest file or by copying an existing Amazon Rekognition Custom Labels dataset.

To create a training dataset for a project, specify `train` for the value of `DatasetType`. To create the test dataset for a project, specify `test` for the value of `DatasetType`.

The response from `CreateDataset` is the Amazon Resource Name (ARN) for the dataset. Creating a dataset takes a while to complete. Use [DescribeDataset \(p. 58\)](#) to check the current status. The dataset created successfully if the value of `Status` is `CREATE_COMPLETE`.

To check if any non-terminal errors occurred, call [ListDatasetEntries \(p. 160\)](#) and check for the presence of `errors` lists in the JSON Lines.

Dataset creation fails if a terminal error occurs (`Status` = `CREATE_FAILED`). Currently, you can't access the terminal error information.

For more information, see [Creating datasets](#).

This operation requires permissions to perform the `rekognition:CreateDataset` action. If you want to copy an existing dataset, you also require permission to perform the `rekognition>ListDatasetEntries` action.

Request Syntax

```
{  
  "DatasetSource": {  
    "DatasetArn": "string",  
    "GroundTruthManifest": {  
      "S3Object": {  
        "Bucket": "string",  
        "Name": "string",  
        "Version": "string"  
      }  
    }  
  },  
  "DatasetType": "string",  
  "ProjectArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

DatasetSource (p. 19)

The source files for the dataset. You can specify the ARN of an existing dataset or specify the Amazon S3 bucket location of an Amazon Sagemaker format manifest file. If you don't specify `datasetSource`, an empty dataset is created. To add labeled images to the dataset, You can use the console or call [UpdateDatasetEntries \(p. 245\)](#).

Type: [DatasetSource \(p. 282\)](#) object

Required: No

[DatasetType \(p. 19\)](#)

The type of the dataset. Specify `train` to create a training dataset. Specify `test` to create a test dataset.

Type: String

Valid Values: TRAIN | TEST

Required: Yes

[ProjectArn \(p. 19\)](#)

The ARN of the Amazon Rekognition Custom Labels project to which you want to assign the dataset.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-zA-Z\d-]+:rekognition:[a-zA-Z\d-]+:\d{12}:project\[/[a-zA-Z0-9_.\-_]{1,255}\/[0-9]+\$)

Required: Yes

Response Syntax

```
{  
  "DatasetArn": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[DatasetArn \(p. 20\)](#)

The ARN of the created Amazon Rekognition Custom Labels dataset.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-zA-Z\d-]+:rekognition:[a-zA-Z\d-]+:\d{12}:project\[/[a-zA-Z0-9_.\-_]{1,255}\/dataset\/(train|test)\/[0-9]+\$)

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a LimitExceededException exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceAlreadyExistsException

A resource with the specified ID already exists.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)

- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

CreateProject

Creates a new Amazon Rekognition Custom Labels project. A project is a group of resources (datasets, model versions) that you use to create and manage Amazon Rekognition Custom Labels models.

This operation requires permissions to perform the `rekognition:CreateProject` action.

Request Syntax

```
{  
  "ProjectName": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

ProjectName (p. 23)

The name of the project to create.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

Response Syntax

```
{  
  "ProjectArn": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ProjectArn (p. 23)

The Amazon Resource Name (ARN) of the new project. You can use the ARN to configure IAM access to the project.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-zA-Z0-9\.\-_]+:rekognition:[a-zA-Z0-9\.\-_]+:\d{12}:project\/[a-zA-Z0-9\.\-_]{1,255}\/[0-9]+\$)

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a LimitExceededException exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)

- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

CreateProjectVersion

Creates a new version of a model and begins training. Models are managed as part of an Amazon Rekognition Custom Labels project. The response from `CreateProjectVersion` is an Amazon Resource Name (ARN) for the version of the model.

Training uses the training and test datasets associated with the project. For more information, see [Creating training and test datasets](#).

Note

You can train a model in a project that doesn't have associated datasets by specifying manifest files in the `TrainingData` and `TestingData` fields.

If you open the console after training a model with manifest files, Amazon Rekognition Custom Labels creates the datasets for you using the most recent manifest files. You can no longer train a model version for the project by specifying manifest files.

Instead of training with a project without associated datasets, we recommend that you use the manifest files to create training and test datasets for the project.

Training takes a while to complete. You can get the current status by calling [DescribeProjectVersions \(p. 64\)](#). Training completed successfully if the value of the `Status` field is `TRAINING_COMPLETED`.

If training fails, see [Debugging a failed model training](#).

Once training has successfully completed, call [DescribeProjectVersions \(p. 64\)](#) to get the training results and evaluate the model. For more information, see [Improving a trained Amazon Rekognition Custom Labels model](#).

After evaluating the model, you start the model by calling [StartProjectVersion \(p. 221\)](#).

This operation requires permissions to perform the `rekognition:CreateProjectVersion` action.

Request Syntax

```
{  
  "KmsKeyId": "string",  
  "OutputConfig": {  
    "S3Bucket": "string",  
    "S3KeyPrefix": "string"  
  },  
  "ProjectArn": "string",  
  "Tags": {  
    "string" : "string"  
  },  
  "TestingData": {  
    "Assets": [  
      {  
        "GroundTruthManifest": {  
          "S3Object": {  
            "Bucket": "string",  
            "Name": "string",  
            "Version": "string"  
          }  
        }  
      }  
    ],  
    "AutoCreate": boolean  
  },  
  "TrainingData": {  
    "Assets": [  
      {  
        "GroundTruthManifest": {  
          "S3Object": {  
            "Bucket": "string",  
            "Name": "string",  
            "Version": "string"  
          }  
        }  
      }  
    ],  
    "AutoCreate": boolean  
  }  
}
```

```
{  
    "GroundTruthManifest": [  
        {"S3Object": {  
            "Bucket": "string",  
            "Name": "string",  
            "Version": "string"  
        }  
    }  
],  
    "VersionName": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[KmsKeyId \(p. 26\)](#)

The identifier for your AWS Key Management Service key (AWS KMS key). You can supply the Amazon Resource Name (ARN) of your KMS key, the ID of your KMS key, an alias for your KMS key, or an alias ARN. The key is used to encrypt training and test images copied into the service for model training. Your source images are unaffected. The key is also used to encrypt training results and manifest files written to the output Amazon S3 bucket (OutputConfig).

If you choose to use your own KMS key, you need the following permissions on the KMS key.

- kms:CreateGrant
- kms:DescribeKey
- kms:GenerateDataKey
- kms:Decrypt

If you don't specify a value for KmsKeyId, images copied into the service are encrypted using a key that AWS owns and manages.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: ^[A-Za-z0-9][A-Za-z0-9:_/+=@.-]{0,2048}\$

Required: No

[OutputConfig \(p. 26\)](#)

The Amazon S3 bucket location to store the results of training. The S3 bucket can be in any AWS account as long as the caller has s3:PutObject permissions on the S3 bucket.

Type: [OutputConfig \(p. 334\)](#) object

Required: Yes

[ProjectArn \(p. 26\)](#)

The ARN of the Amazon Rekognition Custom Labels project that manages the model that you want to train.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+:\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\/[0-9]+\$)

Required: Yes

[Tags \(p. 26\)](#)

A set of tags (key-value pairs) that you want to attach to the model.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 200 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: ^(?!\aws:)[\p{L}\p{Z}\p{N}_.:/=+\-\@]*\$

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: ^([\p{L}\p{Z}\p{N}_.:/=+\-\@]*)\$

Required: No

[TestingData \(p. 26\)](#)

Specifies an external manifest that the service uses to test the model. If you specify TestingData you must also specify TrainingData. The project must not have any associated datasets.

Type: [TestingData \(p. 377\)](#) object

Required: No

[TrainingData \(p. 26\)](#)

Specifies an external manifest that the service uses to train the model. If you specify TrainingData you must also specify TestingData. The project must not have any associated datasets.

Type: [TrainingData \(p. 382\)](#) object

Required: No

[VersionName \(p. 26\)](#)

A name for the version of the model. This value must be unique.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

Response Syntax

```
{  
  "ProjectVersionArn": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ProjectVersionArn (p. 28)

The ARN of the model version that was created. Use `DescribeProjectVersion` to get the current status of the training operation.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-zA-Z\d-]+:rekognition:[a-zA-Z\d-]+\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\version\/[a-zA-Z0-9_.\-\-]{1,255}\[0-9]+\$)

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (`StartLabelDetection`, for example) will raise a `LimitExceededException` exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ServiceQuotaExceededException

The size of the resource exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

CreateStreamProcessor

Creates an Amazon Rekognition stream processor that you can use to detect and recognize faces or to detect labels in a streaming video.

Amazon Rekognition Video is a consumer of live video from Amazon Kinesis Video Streams. There are two different settings for stream processors in Amazon Rekognition: detecting faces and detecting labels.

- If you are creating a stream processor for detecting faces, you provide as input a Kinesis video stream (Input) and a Kinesis data stream (Output) stream. You also specify the face recognition criteria in Settings. For example, the collection containing faces that you want to recognize. After you have finished analyzing a streaming video, use [StopStreamProcessor \(p. 238\)](#) to stop processing.
- If you are creating a stream processor to detect labels, you provide as input a Kinesis video stream (Input), Amazon S3 bucket information (Output), and an Amazon SNS topic ARN (NotificationChannel). You can also provide a KMS key ID to encrypt the data sent to your Amazon S3 bucket. You specify what you want to detect in ConnectedHomeSettings, such as people, packages and people, or pets, people, and packages. You can also specify where in the frame you want Amazon Rekognition to monitor with RegionsOfInterest. When you run the [StartStreamProcessor \(p. 228\)](#) operation on a label detection stream processor, you input start and stop information to determine the length of the processing time.

Use Name to assign an identifier for the stream processor. You use Name to manage the stream processor. For example, you can start processing the source video by calling [StartStreamProcessor \(p. 228\)](#) with the Name field.

This operation requires permissions to perform the `rekognition:CreateStreamProcessor` action. If you want to tag your stream processor, you also require permission to perform the `rekognition:TagResource` operation.

Request Syntax

```
{  
  "DataSharingPreference": {  
    "OptIn": boolean  
  },  
  "Input": {  
    "KinesisVideoStream": {  
      "Arn": string  
    }  
  },  
  "KmsKeyId": string,  
  "Name": string,  
  "NotificationChannel": {  
    "SNSTopicArn": string  
  },  
  "Output": {  
    "KinesisDataStream": {  
      "Arn": string  
    },  
    "S3Destination": {  
      "Bucket": string,  
      "KeyPrefix": string  
    }  
  },  
  "RegionsOfInterest": [  
    {  
      "BoundingBox": {  

```

```
        "Height": number,
        "Left": number,
        "Top": number,
        "Width": number
    },
    "Polygon": [
        {
            "X": number,
            "Y": number
        }
    ]
},
"RoleArn": "string",
"Settings": {
    "ConnectedHome": {
        "Labels": [ "string" ],
        "MinConfidence": number
    },
    "FaceSearch": {
        "CollectionId": "string",
        "FaceMatchThreshold": number
    }
},
"Tags": {
    "string": "string"
}
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[DataSharingPreference \(p. 31\)](#)

Shows whether you are sharing data with Rekognition to improve model performance. You can choose this option at the account level or on a per-stream basis. Note that if you opt out at the account level this setting is ignored on individual streams.

Type: [StreamProcessorDataSharingPreference \(p. 368\)](#) object

Required: No

[Input \(p. 31\)](#)

Kinesis video stream stream that provides the source streaming video. If you are using the AWS CLI, the parameter name is `StreamProcessorInput`. This is required for both face search and label detection stream processors.

Type: [StreamProcessorInput \(p. 369\)](#) object

Required: Yes

[KmsKeyId \(p. 31\)](#)

The identifier for your AWS Key Management Service key (AWS KMS key). This is an optional parameter for label detection stream processors and should not be used to create a face search stream processor. You can supply the Amazon Resource Name (ARN) of your KMS key, the ID of your KMS key, an alias for your KMS key, or an alias ARN. The key is used to encrypt results and data

published to your Amazon S3 bucket, which includes image frames and hero images. Your source images are unaffected.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: ^[A-Za-z0-9][A-Za-z0-9:_/+=@.-]{0,2048}\$

Required: No

[Name \(p. 31\)](#)

An identifier you assign to the stream processor. You can use Name to manage the stream processor. For example, you can get the current status of the stream processor by calling [DescribeStreamProcessor \(p. 69\)](#). Name is idempotent. This is required for both face search and label detection stream processors.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

[NotificationChannel \(p. 31\)](#)

The Amazon Simple Notification Service topic to which Amazon Rekognition publishes the object detection results and completion status of a video analysis operation.

Amazon Rekognition publishes a notification the first time an object of interest or a person is detected in the video stream. For example, if Amazon Rekognition detects a person at second 2, a pet at second 4, and a person again at second 5, Amazon Rekognition sends 2 object class detected notifications, one for a person at second 2 and one for a pet at second 4.

Amazon Rekognition also publishes an end-of-session notification with a summary when the stream processing session is complete.

Type: [StreamProcessorNotificationChannel \(p. 370\)](#) object

Required: No

[Output \(p. 31\)](#)

Kinesis data stream stream or Amazon S3 bucket location to which Amazon Rekognition Video puts the analysis results. If you are using the AWS CLI, the parameter name is StreamProcessorOutput. This must be a [S3Destination \(p. 353\)](#) of an Amazon S3 bucket that you own for a label detection stream processor or a Kinesis data stream ARN for a face search stream processor.

Type: [StreamProcessorOutput \(p. 371\)](#) object

Required: Yes

[RegionsOfInterest \(p. 31\)](#)

Specifies locations in the frames where Amazon Rekognition checks for objects or people. You can specify up to 10 regions of interest, and each region has either a polygon or a bounding box. This is an optional parameter for label detection stream processors and should not be used to create a face search stream processor.

Type: Array of [RegionOfInterest \(p. 352\)](#) objects

Array Members: Minimum number of 0 items. Maximum number of 10 items.

Required: No

[RoleArn \(p. 31\)](#)

The Amazon Resource Number (ARN) of the IAM role that allows access to the stream processor. The IAM role provides Rekognition read permissions for a Kinesis stream. It also provides write permissions to an Amazon S3 bucket and Amazon Simple Notification Service topic for a label detection stream processor. This is required for both face search and label detection stream processors.

Type: String

Pattern: `arn:aws:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\\-/_]+`

Required: Yes

[Settings \(p. 31\)](#)

Input parameters used in a streaming video analyzed by a stream processor. You can use FaceSearch to recognize faces in a streaming video, or you can use ConnectedHome to detect labels.

Type: [StreamProcessorSettings \(p. 372\)](#) object

Required: Yes

[Tags \(p. 31\)](#)

A set of tags (key-value pairs) that you want to attach to the stream processor.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 200 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `^(?!aws:)[\p{L}\p{Z}\p{N}_.:/=+\-@]*$`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

Required: No

Response Syntax

```
{  
  "StreamProcessorArn": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[StreamProcessorArn \(p. 34\)](#)

Amazon Resource Number for the newly created stream processor.

Type: String

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:streamprocessor\/.+)\$)

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a LimitExceededException exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ServiceQuotaExceededException

The size of the resource exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DeleteCollection

Deletes the specified collection. Note that this operation removes all faces in the collection. For an example, see [Deleting a collection](#).

This operation requires permissions to perform the `rekognition:DeleteCollection` action.

Request Syntax

```
{  
  "CollectionId": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

CollectionId (p. 37)

ID of the collection to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\+]

Required: Yes

Response Syntax

```
{  
  "StatusCode": number  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

StatusCode (p. 37)

HTTP status code that indicates the result of the operation.

Type: Integer

Valid Range: Minimum value of 0.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DeleteDataset

Deletes an existing Amazon Rekognition Custom Labels dataset. Deleting a dataset might take while. Use [DescribeDataset \(p. 58\)](#) to check the current status. The dataset is still deleting if the value of Status is DELETE_IN_PROGRESS. If you try to access the dataset after it is deleted, you get a ResourceNotFoundException exception.

You can't delete a dataset while it is creating (Status = CREATE_IN_PROGRESS) or if the dataset is updating (Status = UPDATE_IN_PROGRESS).

This operation requires permissions to perform the `rekognition:DeleteDataset` action.

Request Syntax

```
{  
  "DatasetArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

DatasetArn (p. 39)

The ARN of the Amazon Rekognition Custom Labels dataset that you want to delete.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\dataset\/(train|test)\/[0-9]+\$)

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a LimitExceededException exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DeleteFaces

Deletes faces from a collection. You specify a collection ID and an array of face IDs to remove from the collection.

This operation requires permissions to perform the `rekognition:DeleteFaces` action.

Request Syntax

```
{  
  "CollectionId": "string",  
  "FaceIds": [ "string" ]  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

CollectionId (p. 41)

Collection from which to remove the specific faces.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

FaceIds (p. 41)

An array of face IDs to delete.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4096 items.

Pattern: [0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}

Required: Yes

Response Syntax

```
{  
  "DeletedFaces": [ "string" ]  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[DeletedFaces \(p. 41\)](#)

An array of strings (face IDs) of the faces that were deleted.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4096 items.

Pattern: [0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)

- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DeleteProject

Deletes an Amazon Rekognition Custom Labels project. To delete a project you must first delete all models associated with the project. To delete a model, see [DeleteProjectVersion \(p. 50\)](#).

`DeleteProject` is an asynchronous operation. To check if the project is deleted, call [DescribeProjects \(p. 61\)](#). The project is deleted when the project no longer appears in the response. Be aware that deleting a given project will also delete any `ProjectPolicies` associated with that project.

This operation requires permissions to perform the `rekognition:DeleteProject` action.

Request Syntax

```
{  
  "ProjectArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ProjectArn \(p. 44\)](#)

The Amazon Resource Name (ARN) of the project that you want to delete.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-zA-Z\d-]+:rekognition:[a-zA-Z\d-]+\d{12}:project\/[a-zA-Z0-9_.\-_]{1,255}\/[0-9]+\\$)

Required: Yes

Response Syntax

```
{  
  "Status": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[Status \(p. 44\)](#)

The current status of the delete project operation.

Type: String

Valid Values: CREATING | CREATED | DELETING

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)

- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DeleteProjectPolicy

Deletes an existing project policy.

To get a list of project policies attached to a project, call [ListProjectPolicies \(p. 170\)](#). To attach a project policy to a project, call [PutProjectPolicy \(p. 179\)](#).

Request Syntax

```
{  
  "PolicyName": "string",  
  "PolicyRevisionId": "string",  
  "ProjectArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[PolicyName \(p. 47\)](#)

The name of the policy that you want to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

[PolicyRevisionId \(p. 47\)](#)

The ID of the project policy revision that you want to delete.

Type: String

Length Constraints: Maximum length of 64.

Pattern: [0-9A-Fa-f]+

Required: No

[ProjectArn \(p. 47\)](#)

The Amazon Resource Name (ARN) of the project that the project policy you want to delete is attached to.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-zA-Z0-9\.\-_]+:rekognition:[a-zA-Z0-9\.\-_]+:\d{12}:project\/[a-zA-Z0-9\.\-_]{1,255}\/[0-9]+\$)

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidPolicyRevisionIdException

The supplied revision id for the project policy is invalid.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DeleteProjectVersion

Deletes an Amazon Rekognition Custom Labels model.

You can't delete a model if it is running or if it is training. To check the status of a model, use the [Status](#) field returned from [DescribeProjectVersions \(p. 64\)](#). To stop a running model call [StopProjectVersion \(p. 235\)](#). If the model is training, wait until it finishes.

This operation requires permissions to perform the `rekognition:DeleteProjectVersion` action.

Request Syntax

```
{  
  "ProjectVersionArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ProjectVersionArn \(p. 50\)](#)

The Amazon Resource Name (ARN) of the model version that you want to delete.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `(^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:project\[/[a-zA-Z0-9_.\-_]{1,255}\]/version\[/[a-zA-Z0-9_.\-_]{1,255}\]/[0-9]+$)`

Required: Yes

Response Syntax

```
{  
  "Status": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[Status \(p. 50\)](#)

The status of the deletion operation.

Type: String

Valid Values: TRAINING_IN_PROGRESS | TRAINING_COMPLETED | TRAINING_FAILED | STARTING | RUNNING | FAILED | STOPPING | STOPPED | DELETING | COPYING_IN_PROGRESS | COPYING_COMPLETED | COPYING_FAILED

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DeleteStreamProcessor

Deletes the stream processor identified by Name. You assign the value for Name when you create the stream processor with [CreateStreamProcessor \(p. 31\)](#). You might not be able to use the same name for a stream processor for a few seconds after calling DeleteStreamProcessor.

Request Syntax

```
{  
  "Name": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

Name (p. 53)

The name of the stream processor you want to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DescribeCollection

Describes the specified collection. You can use `DescribeCollection` to get information, such as the number of faces indexed into a collection and the version of the model used by the collection for face detection.

For more information, see [Describing a collection](#).

Request Syntax

```
{  
  "CollectionId": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[CollectionId \(p. 55\)](#)

The ID of the collection to describe.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

Response Syntax

```
{  
  "CollectionARN": "string",  
  "CreationTimestamp": number,  
  "FaceCount": number,  
  "FaceModelVersion": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[CollectionARN \(p. 55\)](#)

The Amazon Resource Name (ARN) of the collection.

Type: String

[CreationTimestamp \(p. 55\)](#)

The number of milliseconds since the Unix epoch time until the creation of the collection. The Unix epoch time is 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970.

Type: Timestamp

[FaceCount \(p. 55\)](#)

The number of faces that are indexed into the collection. To index faces into a collection, use [IndexFaces \(p. 148\)](#).

Type: Long

Valid Range: Minimum value of 0.

[FaceModelVersion \(p. 55\)](#)

The version of the face model that's used by the collection for face detection.

For more information, see [Model versioning](#).

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DescribeDataset

Describes an Amazon Rekognition Custom Labels dataset. You can get information such as the current status of a dataset and statistics about the images and labels in a dataset.

This operation requires permissions to perform the `rekognition:DescribeDataset` action.

Request Syntax

```
{  
  "DatasetArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

DatasetArn (p. 58)

The Amazon Resource Name (ARN) of the dataset that you want to describe.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `(^arn:[a-z\d-]+:rekognition:[a-z\d-]+:\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\dataset\/(train|test)\/[0-9]+$)`

Required: Yes

Response Syntax

```
{  
  "DatasetDescription": {  
    "CreationTimestamp": number,  
    "DatasetStats": {  
      "ErrorEntries": number,  
      "LabeledEntries": number,  
      "TotalEntries": number,  
      "TotalLabels": number  
    },  
    "LastUpdatedTimestamp": number,  
    "Status": "string",  
    "StatusMessage": "string",  
    "StatusMessageCode": "string"  
  }  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

DatasetDescription (p. 58)

The description for the dataset.

Type: [DatasetDescription \(p. 276\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)

- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DescribeProjects

Gets information about your Amazon Rekognition Custom Labels projects.

This operation requires permissions to perform the `rekognition:DescribeProjects` action.

Request Syntax

```
{  
  "MaxResults": number,  
  "NextToken": "string",  
  "ProjectNames": [ "string" ]  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[MaxResults \(p. 61\)](#)

The maximum number of results to return per paginated call. The largest value you can specify is 100. If you specify a value greater than 100, a `ValidationException` error occurs. The default value is 100.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

[NextToken \(p. 61\)](#)

If the previous response was incomplete (because there is more results to retrieve), Amazon Rekognition Custom Labels returns a pagination token in the response. You can use this pagination token to retrieve the next set of results.

Type: String

Length Constraints: Maximum length of 1024.

Required: No

[ProjectNames \(p. 61\)](#)

A list of the projects that you want Amazon Rekognition Custom Labels to describe. If you don't specify a value, the response includes descriptions for all the projects in your AWS account.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_.\-\u00e1-\u00f1]+`

Required: No

Response Syntax

```
{  
    "NextToken": "string",  
    "ProjectDescriptions": [  
        {  
            "CreationTimestamp": number,  
            "Datasets": [  
                {  
                    "CreationTimestamp": number,  
                    "DatasetArn": "string",  
                    "DatasetType": "string",  
                    "Status": "string",  
                    "StatusMessage": "string",  
                    "StatusMessageCode": "string"  
                }  
            ],  
            "ProjectArn": "string",  
            "Status": "string"  
        }  
    ]  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[NextToken \(p. 62\)](#)

If the previous response was incomplete (because there is more results to retrieve), Amazon Rekognition Custom Labels returns a pagination token in the response. You can use this pagination token to retrieve the next set of results.

Type: String

Length Constraints: Maximum length of 1024.

[ProjectDescriptions \(p. 62\)](#)

A list of project descriptions. The list is sorted by the date and time the projects are created.

Type: Array of [ProjectDescription \(p. 341\)](#) objects

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DescribeProjectVersions

Lists and describes the versions of a model in an Amazon Rekognition Custom Labels project. You can specify up to 10 model versions in `ProjectVersionArns`. If you don't specify a value, descriptions for all model versions in the project are returned.

This operation requires permissions to perform the `rekognition:DescribeProjectVersions` action.

Request Syntax

```
{  
    "MaxResults": number,  
    "NextToken": string,  
    "ProjectArn": string,  
    "VersionNames": [ string ]  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

MaxResults (p. 64)

The maximum number of results to return per paginated call. The largest value you can specify is 100. If you specify a value greater than 100, a `ValidationException` error occurs. The default value is 100.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

NextToken (p. 64)

If the previous response was incomplete (because there is more results to retrieve), Amazon Rekognition Custom Labels returns a pagination token in the response. You can use this pagination token to retrieve the next set of results.

Type: String

Length Constraints: Maximum length of 1024.

Required: No

ProjectArn (p. 64)

The Amazon Resource Name (ARN) of the project that contains the models you want to describe.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (`^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:project\[/[a-zA-Z0-9_.\-\d]{1,255}\[/[0-9]+$`)

Required: Yes

VersionNames (p. 64)

A list of model version names that you want to describe. You can add up to 10 model version names to the list. If you don't specify a value, all model descriptions are returned. A version name is part of a model (ProjectVersion) ARN. For example, `my-model.2020-01-21T09.10.15` is the version name in the following ARN. `arn:aws:rekognition:us-east-1:123456789012:project/getting-started/version/my-model.2020-01-21T09.10.15/1234567890123`.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_.\-\+]`

Required: No

Response Syntax

```
{
  "NextToken": "string",
  "ProjectVersionDescriptions": [
    {
      "BillableTrainingTimeInSeconds": number,
      "CreationTimestamp": number,
      "EvaluationResult": {
        "F1Score": number,
        "Summary": {
          "S3Object": {
            "Bucket": "string",
            "Name": "string",
            "Version": "string"
          }
        }
      },
      "KmsKeyId": "string",
      "ManifestSummary": {
        "S3Object": {
          "Bucket": "string",
          "Name": "string",
          "Version": "string"
        }
      },
      "MaxInferenceUnits": number,
      "MinInferenceUnits": number,
      "OutputConfig": {
        "S3Bucket": "string",
        "S3KeyPrefix": "string"
      },
      "ProjectVersionArn": "string",
      "SourceProjectVersionArn": "string",
      "Status": "string",
      "StatusMessage": "string",
      "TestingDataResult": {
        "Input": {
          "Assets": [
            {
              "GroundTruthManifest": {
                "S3Object": {

```

```

        "Bucket": "string",
        "Name": "string",
        "Version": "string"
    }
}
],
"AutoCreate": boolean
},
"Output": {
    "Assets": [
        {
            "GroundTruthManifest": {
                "S3Object": {
                    "Bucket": "string",
                    "Name": "string",
                    "Version": "string"
                }
            }
        ],
        "AutoCreate": boolean
    },
    "Validation": {
        "Assets": [
            {
                "GroundTruthManifest": {
                    "S3Object": {
                        "Bucket": "string",
                        "Name": "string",
                        "Version": "string"
                    }
                }
            }
        ]
    }
},
"TrainingDataResult": {
    "Input": {
        "Assets": [
            {
                "GroundTruthManifest": {
                    "S3Object": {
                        "Bucket": "string",
                        "Name": "string",
                        "Version": "string"
                    }
                }
            }
        ]
    },
    "Output": {
        "Assets": [
            {
                "GroundTruthManifest": {
                    "S3Object": {
                        "Bucket": "string",
                        "Name": "string",
                        "Version": "string"
                    }
                }
            }
        ]
    }
},
"Validation": {
    "Assets": [

```

```
{  
    "GroundTruthManifest": [  
        {"S3Object": {  
            "Bucket": "string",  
            "Name": "string",  
            "Version": "string"  
        }  
    }  
],  
    "TrainingEndTimestamp": number  
}  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[NextToken \(p. 65\)](#)

If the previous response was incomplete (because there is more results to retrieve), Amazon Rekognition Custom Labels returns a pagination token in the response. You can use this pagination token to retrieve the next set of results.

Type: String

Length Constraints: Maximum length of 1024.

[ProjectVersionDescriptions \(p. 65\)](#)

A list of model descriptions. The list is sorted by the creation date and time of the model versions, latest to earliest.

Type: Array of [ProjectVersionDescription \(p. 344\)](#) objects

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidOperationException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DescribeStreamProcessor

Provides information about a stream processor created by [CreateStreamProcessor \(p. 31\)](#). You can get information about the input and output streams, the input parameters for the face recognition being performed, and the current status of the stream processor.

Request Syntax

```
{  
    "Name": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

Name (p. 69)

Name of the stream processor for which you want information.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

Response Syntax

```
{  
    "CreationTimestamp": number,  
    "DataSharingPreference": {  
        "OptIn": boolean  
    },  
    "Input": {  
        "KinesisVideoStream": {  
            "Arn": "string"  
        }  
    },  
    "KmsKeyId": "string",  
    "LastUpdateTimestamp": number,  
    "Name": "string",  
    "NotificationChannel": {  
        "SNSTopicArn": "string"  
    },  
    "Output": {  
        "KinesisDataStream": {  
            "Arn": "string"  
        }  
    },  
    "S3Destination": {  
        "Bucket": "string",  
        "KeyPrefix": "string"  
    }  
}
```

```

        },
        "RegionsOfInterest": [
        {
            "BoundingBox": {
                "Height": number,
                "Left": number,
                "Top": number,
                "Width": number
            },
            "Polygon": [
            {
                "X": number,
                "Y": number
            }
            ]
        }
    ],
    "RoleArn": "string",
    "Settings": {
        "ConnectedHome": {
            "Labels": [ "string" ],
            "MinConfidence": number
        },
        "FaceSearch": {
            "CollectionId": "string",
            "FaceMatchThreshold": number
        }
    },
    "Status": "string",
    "StatusMessage": "string",
    "StreamProcessorArn": "string"
}

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[CreationTimestamp \(p. 69\)](#)

Date and time the stream processor was created

Type: Timestamp

[DataSharingPreference \(p. 69\)](#)

Shows whether you are sharing data with Rekognition to improve model performance. You can choose this option at the account level or on a per-stream basis. Note that if you opt out at the account level this setting is ignored on individual streams.

Type: [StreamProcessorDataSharingPreference \(p. 368\)](#) object

[Input \(p. 69\)](#)

Kinesis video stream that provides the source streaming video.

Type: [StreamProcessorInput \(p. 369\)](#) object

[KmsKeyId \(p. 69\)](#)

The identifier for your AWS Key Management Service key (AWS KMS key). This is an optional parameter for label detection stream processors.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: ^[A-Za-z0-9][A-Za-z0-9:_/+,@.-]{0,2048}\$

[LastUpdateTimestamp \(p. 69\)](#)

The time, in Unix format, the stream processor was last updated. For example, when the stream processor moves from a running state to a failed state, or when the user starts or stops the stream processor.

Type: Timestamp

[Name \(p. 69\)](#)

Name of the stream processor.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: [a-zA-Z0-9_.\-\+]+

[NotificationChannel \(p. 69\)](#)

The Amazon Simple Notification Service topic to which Amazon Rekognition publishes the object detection results and completion status of a video analysis operation.

Amazon Rekognition publishes a notification the first time an object of interest or a person is detected in the video stream. For example, if Amazon Rekognition detects a person at second 2, a pet at second 4, and a person again at second 5, Amazon Rekognition sends 2 object class detected notifications, one for a person at second 2 and one for a pet at second 4.

Amazon Rekognition also publishes an end-of-session notification with a summary when the stream processing session is complete.

Type: [StreamProcessorNotificationChannel \(p. 370\)](#) object

[Output \(p. 69\)](#)

Kinesis data stream to which Amazon Rekognition Video puts the analysis results.

Type: [StreamProcessorOutput \(p. 371\)](#) object

[RegionsOfInterest \(p. 69\)](#)

Specifies locations in the frames where Amazon Rekognition checks for objects or people. This is an optional parameter for label detection stream processors.

Type: Array of [RegionOfInterest \(p. 352\)](#) objects

Array Members: Minimum number of 0 items. Maximum number of 10 items.

[RoleArn \(p. 69\)](#)

ARN of the IAM role that allows access to the stream processor.

Type: String

Pattern: arn:aws:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/\]+

[Settings \(p. 69\)](#)

Input parameters used in a streaming video analyzed by a stream processor. You can use FaceSearch to recognize faces in a streaming video, or you can use ConnectedHome to detect labels.

Type: [StreamProcessorSettings \(p. 372\)](#) object

Status (p. 69)

Current status of the stream processor.

Type: String

Valid Values: STOPPED | STARTING | RUNNING | FAILED | STOPPING | UPDATING

StatusMessage (p. 69)

Detailed status message about the stream processor.

Type: String

StreamProcessorArn (p. 69)

ARN of the stream processor.

Type: String

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:streamprocessor\/.+)\$)

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DetectCustomLabels

Detects custom labels in a supplied image by using an Amazon Rekognition Custom Labels model.

You specify which version of a model version to use by using the `ProjectVersionArn` input parameter.

You pass the input image as base64-encoded image bytes or as a reference to an image in an Amazon S3 bucket. If you use the AWS CLI to call Amazon Rekognition operations, passing image bytes is not supported. The image must be either a PNG or JPEG formatted file.

For each object that the model version detects on an image, the API returns a (`CustomLabel`) object in an array (`CustomLabels`). Each `CustomLabel` object provides the label name (`Name`), the level of confidence that the image contains the object (`Confidence`), and object location information, if it exists, for the label on the image (`Geometry`).

To filter labels that are returned, specify a value for `MinConfidence`. `DetectCustomLabels` only returns labels with a confidence that's higher than the specified value. The value of `MinConfidence` maps to the assumed threshold values created during training. For more information, see [Assumed Threshold](#). Amazon Rekognition Custom Labels expresses an assumed threshold as a floating point value between 0-1. The range of `MinConfidence` normalizes the assumed threshold to a percentage value (0-100). Confidence responses from `DetectCustomLabels` are also returned as a percentage. You can use `MinConfidence` to change the precision and recall of your model. For more information, see [Analyzing an image](#).

If you don't specify a value for `MinConfidence`, `DetectCustomLabels` returns labels based on the assumed threshold of each label.

This is a stateless API operation. That is, the operation does not persist any data.

This operation requires permissions to perform the `rekognition:DetectCustomLabels` action.

For more information, see [Analyzing an image](#).

Request Syntax

```
{  
  "Image": {  
    "Bytes": blob,  
    "S3Object": {  
      "Bucket": "string",  
      "Name": "string",  
      "Version": "string"  
    }  
  },  
  "MaxResults": number,  
  "MinConfidence": number,  
  "ProjectVersionArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

Image (p. 74)

Provides the input image either as bytes or an S3 object.

You pass image bytes to an Amazon Rekognition API operation by using the Bytes property. For example, you would use the Bytes property to pass an image loaded from a local file system. Image bytes passed by using the Bytes property must be base64-encoded. Your code may not need to encode image bytes if you are using an AWS SDK to call Amazon Rekognition API operations.

For more information, see [Analyzing an image loaded from a local file system](#).

You pass images stored in an S3 bucket to an Amazon Rekognition API operation by using the S3Object property. Images stored in an S3 bucket do not need to be base64-encoded.

The region for the S3 bucket containing the S3 object must match the region you use for Amazon Rekognition operations.

If you use the AWS CLI to call Amazon Rekognition operations, passing image bytes using the Bytes property is not supported. You must first upload the image to an Amazon S3 bucket and then call the operation using the S3Object property.

For Amazon Rekognition to process an S3 object, the user must have permission to access the S3 object. For more information, see [How Amazon Rekognition works with IAM](#).

Type: [Image \(p. 317\)](#) object

Required: Yes

MaxResults (p. 74)

Maximum number of results you want the service to return in the response. The service returns the specified number of highest confidence labels ranked from highest confidence to lowest.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

MinConfidence (p. 74)

Specifies the minimum confidence level for the labels to return. DetectCustomLabels doesn't return any labels with a confidence value that's lower than this specified value. If you specify a value of 0, DetectCustomLabels returns all labels, regardless of the assumed threshold applied to each label. If you don't specify a value for MinConfidence, DetectCustomLabels returns labels based on the assumed threshold of each label.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

ProjectVersionArn (p. 74)

The ARN of the model version that you want to use.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\:\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\version\/[a-zA-Z0-9_.\-\-]{1,255}\/[0-9]+\\$)

Required: Yes

Response Syntax

```
{  
  "CustomLabels": [  
    {  
      "Confidence": number,  
      "Geometry": {  
        "BoundingBox": {  
          "Height": number,  
          "Left": number,  
          "Top": number,  
          "Width": number  
        },  
        "Polygon": [  
          {  
            "X": number,  
            "Y": number  
          }  
        ]  
      },  
      "Name": "string"  
    }  
  ]  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

CustomLabels (p. 76)

An array of custom labels detected in the input image.

Type: Array of [CustomLabel \(p. 274\)](#) objects

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

ImageTooLargeException

The input image size exceeds the allowed limit. If you are calling [DetectProtectiveEquipment \(p. 95\)](#), the image size or resolution exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidImageFormatException

The provided image format is not supported.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a LimitExceededException exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ResourceNotReadyException

The requested resource isn't ready. For example, this exception occurs when you call DetectCustomLabels with a model version that isn't deployed.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DetectFaces

Detects faces within an image that is provided as input.

DetectFaces detects the 100 largest faces in the image. For each face detected, the operation returns face details. These details include a bounding box of the face, a confidence value (that the bounding box contains a face), and a fixed set of attributes such as facial landmarks (for example, coordinates of eye and mouth), presence of beard, sunglasses, and so on.

The face-detection algorithm is most effective on frontal faces. For non-frontal or obscured faces, the algorithm might not detect the faces or might detect faces with lower confidence.

You pass the input image either as base64-encoded image bytes or as a reference to an image in an Amazon S3 bucket. If you use the AWS CLI to call Amazon Rekognition operations, passing image bytes is not supported. The image must be either a PNG or JPEG formatted file.

Note

This is a stateless API operation. That is, the operation does not persist any data.

This operation requires permissions to perform the `rekognition:DetectFaces` action.

Request Syntax

```
{  
  "Attributes": [ "string" ],  
  "Image": {  
    "Bytes": blob,  
    "S3Object": {  
      "Bucket": "string",  
      "Name": "string",  
      "Version": "string"  
    }  
  }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

Attributes (p. 79)

An array of facial attributes you want to be returned. This can be the default list of attributes or all attributes. If you don't specify a value for Attributes or if you specify `["DEFAULT"]`, the API returns the following subset of facial attributes: BoundingBox, Confidence, Pose, Quality, and Landmarks. If you provide `["ALL"]`, all facial attributes are returned, but the operation takes longer to complete.

If you provide both, `["ALL", "DEFAULT"]`, the service uses a logical AND operator to determine which attributes to return (in this case, all attributes).

Type: Array of strings

Valid Values: DEFAULT | ALL

Required: No

Image (p. 79)

The input image as base64-encoded bytes or an S3 object. If you use the AWS CLI to call Amazon Rekognition operations, passing base64-encoded image bytes is not supported.

If you are using an AWS SDK to call Amazon Rekognition, you might not need to base64-encode image bytes passed using the Bytes field. For more information, see [Image specifications](#).

Type: [Image \(p. 317\)](#) object

Required: Yes

Response Syntax

```
{  
  "FaceDetails": [  
    {  
      "AgeRange": {  
        "High": number,  
        "Low": number  
      },  
      "Beard": {  
        "Confidence": number,  
        "Value": boolean  
      },  
      "BoundingBox": {  
        "Height": number,  
        "Left": number,  
        "Top": number,  
        "Width": number  
      },  
      "Confidence": number,  
      "Emotions": [  
        {  
          "Confidence": number,  
          "Type": "string"  
        }  
      ],  
      "Eyeglasses": {  
        "Confidence": number,  
        "Value": boolean  
      },  
      "EyesOpen": {  
        "Confidence": number,  
        "Value": boolean  
      },  
      "Gender": {  
        "Confidence": number,  
        "Value": "string"  
      },  
      "Landmarks": [  
        {  
          "Type": "string",  
          "X": number,  
          "Y": number  
        }  
      ],  
      "MouthOpen": {  
        "Confidence": number,  
        "Value": boolean  
      },  
      "Mustache": {  
        "Confidence": number,  
        "Value": boolean  
      }  
    }  
  ]  
}
```

```

        "Value": boolean
    },
    "Pose": {
        "Pitch": number,
        "Roll": number,
        "Yaw": number
    },
    "Quality": {
        "Brightness": number,
        "Sharpness": number
    },
    "Smile": {
        "Confidence": number,
        "Value": boolean
    },
    "Sunglasses": {
        "Confidence": number,
        "Value": boolean
    }
}
],
"OrientationCorrection": "string"
}

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[FaceDetails \(p. 80\)](#)

Details of each face found in the image.

Type: Array of [FaceDetail \(p. 302\)](#) objects

[OrientationCorrection \(p. 80\)](#)

The value of OrientationCorrection is always null.

If the input image is in .jpeg format, it might contain exchangeable image file format (Exif) metadata that includes the image's orientation. Amazon Rekognition uses this orientation information to perform image correction. The bounding box coordinates are translated to represent object locations after the orientation information in the Exif metadata is used to correct the image orientation. Images in .png format don't contain Exif metadata.

Amazon Rekognition doesn't perform image correction for images in .png format and .jpeg images without orientation information in the image Exif metadata. The bounding box coordinates aren't translated and represent the object locations before the image is rotated.

Type: String

Valid Values: ROTATE_0 | ROTATE_90 | ROTATE_180 | ROTATE_270

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

[AccessDeniedException](#)

You are not authorized to perform the action.

HTTP Status Code: 400

ImageTooLargeException

The input image size exceeds the allowed limit. If you are calling [DetectProtectiveEquipment \(p. 95\)](#), the image size or resolution exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidImageFormatException

The provided image format is not supported.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DetectLabels

Detects instances of real-world entities within an image (JPEG or PNG) provided as input. This includes objects like flower, tree, and table; events like wedding, graduation, and birthday party; and concepts like landscape, evening, and nature.

For an example, see [Analyzing images stored in an Amazon S3 bucket](#).

You pass the input image as base64-encoded image bytes or as a reference to an image in an Amazon S3 bucket. If you use the AWS CLI to call Amazon Rekognition operations, passing image bytes is not supported. The image must be either a PNG or JPEG formatted file.

Optional Parameters

You can specify one or both of the GENERAL_LABELS and IMAGE_PROPERTIES feature types when calling the DetectLabels API. Including GENERAL_LABELS will ensure the response includes the labels detected in the input image, while including IMAGE_PROPERTIES will ensure the response includes information about the image quality and color.

When using GENERAL_LABELS and/or IMAGE_PROPERTIES you can provide filtering criteria to the Settings parameter. You can filter with sets of individual labels or with label categories. You can specify inclusive filters, exclusive filters, or a combination of inclusive and exclusive filters. For more information on filtering see [Detecting Labels in an Image](#).

You can specify MinConfidence to control the confidence threshold for the labels returned. The default is 55%. You can also add the MaxLabels parameter to limit the number of labels returned. The default and upper limit is 1000 labels.

Response Elements

For each object, scene, and concept the API returns one or more labels. The API returns the following types of information regarding labels:

- Name - The name of the detected label.
- Confidence - The level of confidence in the label assigned to a detected object.
- Parents - The ancestor labels for a detected label. DetectLabels returns a hierarchical taxonomy of detected labels. For example, a detected car might be assigned the label car. The label car has two parent labels: Vehicle (its parent) and Transportation (its grandparent). The response includes the all ancestors for a label, where every ancestor is a unique label. In the previous example, Car, Vehicle, and Transportation are returned as unique labels in the response.
- Aliases - Possible Aliases for the label.
- Categories - The label categories that the detected label belongs to.
- BoundingBox — Bounding boxes are described for all instances of detected common object labels, returned in an array of Instance objects. An Instance object contains a BoundingBox object, describing the location of the label on the input image. It also includes the confidence for the accuracy of the detected bounding box.

The API returns the following information regarding the image, as part of the ImageProperties structure:

- Quality - Information about the Sharpness, Brightness, and Contrast of the input image, scored between 0 to 100. Image quality is returned for the entire image, as well as the background and the foreground.
- Dominant Color - An array of the dominant colors in the image.
- Foreground - Information about the sharpness, brightness, and dominant colors of the input image's foreground.

- Background - Information about the sharpness, brightness, and dominant colors of the input image's background.

The list of returned labels will include at least one label for every detected object, along with information about that label. In the following example, suppose the input image has a lighthouse, the sea, and a rock. The response includes all three labels, one for each object, as well as the confidence in the label:

```
{Name: lighthouse, Confidence: 98.4629}

{Name: rock, Confidence: 79.2097}

{Name: sea, Confidence: 75.061}
```

The list of labels can include multiple labels for the same object. For example, if the input image shows a flower (for example, a tulip), the operation might return the following three labels.

```
{Name: flower, Confidence: 99.0562}

{Name: plant, Confidence: 99.0562}

{Name: tulip, Confidence: 99.0562}
```

In this example, the detection algorithm more precisely identifies the flower as a tulip.

Note

If the object detected is a person, the operation doesn't provide the same facial details that the [DetectFaces \(p. 79\)](#) operation provides.

This is a stateless API operation. That is, the operation does not persist any data.

This operation requires permissions to perform the `rekognition:DetectLabels` action.

Request Syntax

```
{
  "Features": [ "string" ],
  "Image": {
    "Bytes": blob,
    "S3Object": {
      "Bucket": "string",
      "Name": "string",
      "Version": "string"
    }
  },
  "MaxLabels": number,
  "MinConfidence": number,
  "Settings": {
    "GeneralLabels": {
      "LabelCategoryExclusionFilters": [ "string" ],
      "LabelCategoryInclusionFilters": [ "string" ],
      "LabelExclusionFilters": [ "string" ],
      "LabelInclusionFilters": [ "string" ]
    },
    "ImageProperties": {
      "MaxDominantColors": number
    }
  }
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

Features (p. 85)

A list of the types of analysis to perform. Specifying GENERAL_LABELS uses the label detection feature, while specifying IMAGE_PROPERTIES returns information regarding image color and quality. If no option is specified GENERAL_LABELS is used by default.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 2 items.

Valid Values: GENERAL_LABELS | IMAGE_PROPERTIES

Required: No

Image (p. 85)

The input image as base64-encoded bytes or an S3 object. If you use the AWS CLI to call Amazon Rekognition operations, passing image bytes is not supported. Images stored in an S3 Bucket do not need to be base64-encoded.

If you are using an AWS SDK to call Amazon Rekognition, you might not need to base64-encode image bytes passed using the Bytes field. For more information, see [Image specifications](#).

Type: [Image \(p. 317\)](#) object

Required: Yes

MaxLabels (p. 85)

Maximum number of labels you want the service to return in the response. The service returns the specified number of highest confidence labels.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

MinConfidence (p. 85)

Specifies the minimum confidence level for the labels to return. Amazon Rekognition doesn't return any labels with confidence lower than this specified value.

If MinConfidence is not specified, the operation returns labels with a confidence values greater than or equal to 55 percent.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Settings (p. 85)

A list of the filters to be applied to returned detected labels and image properties. Specified filters can be inclusive, exclusive, or a combination of both. Filters can be used for individual labels or label

categories. The exact label names or label categories must be supplied. For a full list of labels and label categories, see [LINK HERE](#).

Type: [DetectLabelsSettings \(p. 290\)](#) object

Required: No

Response Syntax

```
{
  "ImageProperties": {
    "Background": {
      "DominantColors": [
        {
          "Blue": number,
          "CSSColor": "string",
          "Green": number,
          "HexCode": "string",
          "PixelPercent": number,
          "Red": number,
          "SimplifiedColor": "string"
        }
      ],
      "Quality": {
        "Brightness": number,
        "Contrast": number,
        "Sharpness": number
      }
    },
    "DominantColors": [
      {
        "Blue": number,
        "CSSColor": "string",
        "Green": number,
        "HexCode": "string",
        "PixelPercent": number,
        "Red": number,
        "SimplifiedColor": "string"
      }
    ],
    "Foreground": {
      "DominantColors": [
        {
          "Blue": number,
          "CSSColor": "string",
          "Green": number,
          "HexCode": "string",
          "PixelPercent": number,
          "Red": number,
          "SimplifiedColor": "string"
        }
      ],
      "Quality": {
        "Brightness": number,
        "Contrast": number,
        "Sharpness": number
      }
    },
    "Quality": {
      "Brightness": number,
      "Contrast": number,
      "Sharpness": number
    }
  }
}
```

```

        },
        "LabelModelVersion": "string",
        "Labels": [
            {
                "Aliases": [
                    {
                        "Name": "string"
                    }
                ],
                "Categories": [
                    {
                        "Name": "string"
                    }
                ],
                "Confidence": number,
                "Instances": [
                    {
                        "BoundingBox": {
                            "Height": number,
                            "Left": number,
                            "Top": number,
                            "Width": number
                        },
                        "Confidence": number,
                        "DominantColors": [
                            {
                                "Blue": number,
                                "CSSColor": "string",
                                "Green": number,
                                "HexCode": "string",
                                "PixelPercent": number,
                                "Red": number,
                                "SimplifiedColor": "string"
                            }
                        ]
                    }
                ],
                "Name": "string",
                "Parents": [
                    {
                        "Name": "string"
                    }
                ]
            }
        ],
        "OrientationCorrection": "string"
    }
}

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[ImageProperties \(p. 87\)](#)

Information about the properties of the input image, such as brightness, sharpness, contrast, and dominant colors.

Type: [DetectLabelsImageProperties \(p. 287\)](#) object

[LabelModelVersion \(p. 87\)](#)

Version number of the label detection model that was used to detect labels.

Type: String

Labels (p. 87)

An array of labels for the real-world objects detected.

Type: Array of [Label \(p. 324\)](#) objects

OrientationCorrection (p. 87)

The value of OrientationCorrection is always null.

If the input image is in .jpeg format, it might contain exchangeable image file format (Exif) metadata that includes the image's orientation. Amazon Rekognition uses this orientation information to perform image correction. The bounding box coordinates are translated to represent object locations after the orientation information in the Exif metadata is used to correct the image orientation. Images in .png format don't contain Exif metadata.

Amazon Rekognition doesn't perform image correction for images in .png format and .jpeg images without orientation information in the image Exif metadata. The bounding box coordinates aren't translated and represent the object locations before the image is rotated.

Type: String

Valid Values: ROTATE_0 | ROTATE_90 | ROTATE_180 | ROTATE_270

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

ImageTooLargeException

The input image size exceeds the allowed limit. If you are calling [DetectProtectiveEquipment \(p. 95\)](#), the image size or resolution exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidImageFormatException

The provided image format is not supported.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DetectModerationLabels

Detects unsafe content in a specified JPEG or PNG format image. Use `DetectModerationLabels` to moderate images depending on your requirements. For example, you might want to filter images that contain nudity, but not images containing suggestive content.

To filter images, use the labels returned by `DetectModerationLabels` to determine which types of content are appropriate.

For information about moderation labels, see [Moderating content](#). For a list of moderation labels in Amazon Rekognition, see [Using the image and video moderation APIs](#).

You pass the input image either as base64-encoded image bytes or as a reference to an image in an Amazon S3 bucket. If you use the AWS CLI to call Amazon Rekognition operations, passing image bytes is not supported. The image must be either a PNG or JPEG formatted file.

Request Syntax

```
{  
  "HumanLoopConfig": {  
    "DataAttributes": {  
      "ContentClassifiers": [ "string" ]  
    },  
    "FlowDefinitionArn": "string",  
    "HumanLoopName": "string"  
  },  
  "Image": {  
    "Bytes": blob,  
    "S3Object": {  
      "Bucket": "string",  
      "Name": "string",  
      "Version": "string"  
    }  
  },  
  "MinConfidence": number  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[HumanLoopConfig \(p. 91\)](#)

Sets up the configuration for human evaluation, including the FlowDefinition the image will be sent to.

Type: [HumanLoopConfig \(p. 315\)](#) object

Required: No

[Image \(p. 91\)](#)

The input image as base64-encoded bytes or an S3 object. If you use the AWS CLI to call Amazon Rekognition operations, passing base64-encoded image bytes is not supported.

If you are using an AWS SDK to call Amazon Rekognition, you might not need to base64-encode image bytes passed using the `Bytes` field. For more information, see [Image specifications](#).

Type: [Image \(p. 317\)](#) object

Required: Yes

[MinConfidence \(p. 91\)](#)

Specifies the minimum confidence level for the labels to return. Amazon Rekognition doesn't return any labels with a confidence level lower than this specified value.

If you don't specify `MinConfidence`, the operation returns labels with confidence values greater than or equal to 50 percent.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Response Syntax

```
{  
    "HumanLoopActivationOutput": {  
        "HumanLoopActivationConditionsEvaluationResults": "string",  
        "HumanLoopActivationReasons": [ "string" ],  
        "HumanLoopArn": "string"  
    },  
    "ModerationLabels": [  
        {  
            "Confidence": number,  
            "Name": "string",  
            "ParentName": "string"  
        }  
    ],  
    "ModerationModelVersion": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[HumanLoopActivationOutput \(p. 92\)](#)

Shows the results of the human in the loop evaluation.

Type: [HumanLoopActivationOutput \(p. 314\)](#) object

[ModerationLabels \(p. 92\)](#)

Array of detected Moderation labels and the time, in milliseconds from the start of the video, they were detected.

Type: Array of [ModerationLabel \(p. 330\)](#) objects

[ModerationModelVersion \(p. 92\)](#)

Version number of the moderation detection model that was used to detect unsafe content.

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

HumanLoopQuotaExceededException

The number of in-progress human reviews you have has exceeded the number allowed.

HTTP Status Code: 400

ImageTooLargeException

The input image size exceeds the allowed limit. If you are calling [DetectProtectiveEquipment \(p. 95\)](#), the image size or resolution exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidImageFormatException

The provided image format is not supported.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DetectProtectiveEquipment

Detects Personal Protective Equipment (PPE) worn by people detected in an image. Amazon Rekognition can detect the following types of PPE.

- Face cover
- Hand cover
- Head cover

You pass the input image as base64-encoded image bytes or as a reference to an image in an Amazon S3 bucket. The image must be either a PNG or JPG formatted file.

`DetectProtectiveEquipment` detects PPE worn by up to 15 persons detected in an image.

For each person detected in the image the API returns an array of body parts (face, head, left-hand, right-hand). For each body part, an array of detected items of PPE is returned, including an indicator of whether or not the PPE covers the body part. The API returns the confidence it has in each detection (person, PPE, body part and body part coverage). It also returns a bounding box ([BoundingBox \(p. 259\)](#)) for each detected person and each detected item of PPE.

You can optionally request a summary of detected PPE items with the `SummarizationAttributes` input parameter. The summary provides the following information.

- The persons detected as wearing all of the types of PPE that you specify.
- The persons detected as not wearing all of the types PPE that you specify.
- The persons detected where PPE adornment could not be determined.

This is a stateless API operation. That is, the operation does not persist any data.

This operation requires permissions to perform the `rekognition:DetectProtectiveEquipment` action.

Request Syntax

```
{  
  "Image": {  
    "Bytes": blob,  
    "S3Object": {  
      "Bucket": "string",  
      "Name": "string",  
      "Version": "string"  
    }  
  },  
  "SummarizationAttributes": {  
    "MinConfidence": number,  
    "RequiredEquipmentTypes": [ "string" ]  
  }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

Image (p. 95)

The image in which you want to detect PPE on detected persons. The image can be passed as image bytes or you can reference an image stored in an Amazon S3 bucket.

Type: [Image \(p. 317\)](#) object

Required: Yes

SummarizationAttributes (p. 95)

An array of PPE types that you want to summarize.

Type: [ProtectiveEquipmentSummarizationAttributes \(p. 349\)](#) object

Required: No

Response Syntax

```
{
  "Persons": [
    {
      "BodyParts": [
        {
          "Confidence": number,
          "EquipmentDetections": [
            {
              "BoundingBox": {
                "Height": number,
                "Left": number,
                "Top": number,
                "Width": number
              },
              "Confidence": number,
              "CoversBodyPart": {
                "Confidence": number,
                "Value": boolean
              },
              "Type": "string"
            }
          ],
          "Name": "string"
        }
      ],
      "BoundingBox": {
        "Height": number,
        "Left": number,
        "Top": number,
        "Width": number
      },
      "Confidence": number,
      "Id": number
    }
  ],
  "ProtectiveEquipmentModelVersion": "string",
  "Summary": {
    "PersonsIndeterminate": [ number ],
    "PersonsWithoutRequiredEquipment": [ number ],
    "PersonsWithRequiredEquipment": [ number ]
  }
}
```

}

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[Persons \(p. 96\)](#)

An array of persons detected in the image (including persons not wearing PPE).

Type: Array of [ProtectiveEquipmentPerson \(p. 348\)](#) objects

[ProtectiveEquipmentModelVersion \(p. 96\)](#)

The version number of the PPE detection model used to detect PPE in the image.

Type: String

[Summary \(p. 96\)](#)

Summary information for the types of PPE specified in the SummarizationAttributes input parameter.

Type: [ProtectiveEquipmentSummary \(p. 350\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

ImageTooLargeException

The input image size exceeds the allowed limit. If you are calling [DetectProtectiveEquipment \(p. 95\)](#), the image size or resolution exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidImageFormatException

The provided image format is not supported.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DetectText

Detects text in the input image and converts it into machine-readable text.

Pass the input image as base64-encoded image bytes or as a reference to an image in an Amazon S3 bucket. If you use the AWS CLI to call Amazon Rekognition operations, you must pass it as a reference to an image in an Amazon S3 bucket. For the AWS CLI, passing image bytes is not supported. The image must be either a .png or .jpeg formatted file.

The DetectText operation returns text in an array of [TextDetection \(p. 379\)](#) elements, TextDetections. Each TextDetection element provides information about a single word or line of text that was detected in the image.

A word is one or more script characters that are not separated by spaces. DetectText can detect up to 100 words in an image.

A line is a string of equally spaced words. A line isn't necessarily a complete sentence. For example, a driver's license number is detected as a line. A line ends when there is no aligned text after it. Also, a line ends when there is a large gap between words, relative to the length of the words. This means, depending on the gap between words, Amazon Rekognition may detect multiple lines in text aligned in the same direction. Periods don't represent the end of a line. If a sentence spans multiple lines, the DetectText operation returns multiple lines.

To determine whether a TextDetection element is a line of text or a word, use the TextDetection object Type field.

To be detected, text must be within +/- 90 degrees orientation of the horizontal axis.

For more information, see [Detecting text](#).

Request Syntax

```
{  
  "Filters": {  
    "RegionsOfInterest": [  
      {  
        "BoundingBox": {  
          "Height": number,  
          "Left": number,  
          "Top": number,  
          "Width": number  
        },  
        "Polygon": [  
          {  
            "X": number,  
            "Y": number  
          }  
        ]  
      }  
    ],  
    "WordFilter": {  
      "MinBoundingBoxHeight": number,  
      "MinBoundingBoxWidth": number,  
      "MinConfidence": number  
    }  
  },  
  "Image": {  
    "Bytes": blob,  
    "S3Object": {  
      "Bucket": "string",  
      "Name": "string"  
    }  
  }  
}
```

```
        "Name": "string",
        "Version": "string"
    }
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[Filters \(p. 99\)](#)

Optional parameters that let you set the criteria that the text must meet to be included in your response.

Type: [DetectTextFilters \(p. 291\)](#) object

Required: No

[Image \(p. 99\)](#)

The input image as base64-encoded bytes or an Amazon S3 object. If you use the AWS CLI to call Amazon Rekognition operations, you can't pass image bytes.

If you are using an AWS SDK to call Amazon Rekognition, you might not need to base64-encode image bytes passed using the Bytes field. For more information, see [Image specifications](#).

Type: [Image \(p. 317\)](#) object

Required: Yes

Response Syntax

```
{
    "TextDetections": [
        {
            "Confidence": number,
            "DetectedText": "string",
            "Geometry": {
                "BoundingBox": {
                    "Height": number,
                    "Left": number,
                    "Top": number,
                    "Width": number
                },
                "Polygon": [
                    {
                        "X": number,
                        "Y": number
                    }
                ]
            },
            "Id": number,
            "ParentId": number,
            "Type": "string"
        }
    ],
}
```

```
    "TextModelVersion": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[TextDetections \(p. 100\)](#)

An array of text that was detected in the input image.

Type: Array of [TextDetection \(p. 379\)](#) objects

[TextModelVersion \(p. 100\)](#)

The model version used to detect text.

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

ImageTooLargeException

The input image size exceeds the allowed limit. If you are calling [DetectProtectiveEquipment \(p. 95\)](#), the image size or resolution exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidImageFormatException

The provided image format is not supported.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DistributeDatasetEntries

Distributes the entries (images) in a training dataset across the training dataset and the test dataset for a project. `DistributeDatasetEntries` moves 20% of the training dataset images to the test dataset. An entry is a JSON Line that describes an image.

You supply the Amazon Resource Names (ARN) of a project's training dataset and test dataset. The training dataset must contain the images that you want to split. The test dataset must be empty. The datasets must belong to the same project. To create training and test datasets for a project, call [CreateDataset \(p. 19\)](#).

Distributing a dataset takes a while to complete. To check the status call `DescribeDataset`. The operation is complete when the `Status` field for the training dataset and the test dataset is `UPDATE_COMPLETE`. If the dataset split fails, the value of `Status` is `UPDATE_FAILED`.

This operation requires permissions to perform the `rekognition:DistributeDatasetEntries` action.

Request Syntax

```
{  
  "Datasets": [  
    {  
      "Arn": "string"  
    }  
  ]  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

Datasets (p. 103)

The ARNs for the training dataset and test dataset that you want to use. The datasets must belong to the same project. The test dataset must be empty.

Type: Array of [DistributeDataset \(p. 292\)](#) objects

Array Members: Fixed number of 2 items.

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ResourceNotReadyException

The requested resource isn't ready. For example, this exception occurs when you call DetectCustomLabels with a model version that isn't deployed.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetCelebrityInfo

Gets the name and additional information about a celebrity based on their Amazon Rekognition ID. The additional information is returned as an array of URLs. If there is no additional information about the celebrity, this list is empty.

For more information, see [Getting information about a celebrity](#).

This operation requires permissions to perform the `rekognition:GetCelebrityInfo` action.

Request Syntax

```
{  
    "Id": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

Id (p. 105)

The ID for the celebrity. You get the celebrity ID from a call to the [RecognizeCelebrities \(p. 183\)](#) operation, which recognizes celebrities in an image.

Type: String

Pattern: [0-9A-Za-z]*

Required: Yes

Response Syntax

```
{  
    "KnownGender": {  
        "Type": "string"  
    },  
    "Name": "string",  
    "Urls": [ "string" ]  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

KnownGender (p. 105)

Retrieves the known gender for the celebrity.

Type: [KnownGender \(p. 323\)](#) object

[Name \(p. 105\)](#)

The name of the celebrity.

Type: String

[Urls \(p. 105\)](#)

An array of URLs pointing to additional celebrity information.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 255 items.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetCelebrityRecognition

Gets the celebrity recognition results for a Amazon Rekognition Video analysis started by [StartCelebrityRecognition \(p. 197\)](#).

Celebrity recognition in a video is an asynchronous operation. Analysis is started by a call to [StartCelebrityRecognition \(p. 197\)](#) which returns a job identifier (JobId).

When the celebrity recognition operation finishes, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic registered in the initial call to [StartCelebrityRecognition](#). To get the results of the celebrity recognition analysis, first check that the status value published to the Amazon SNS topic is SUCCEEDED. If so, call [GetCelebrityDetection](#) and pass the job identifier (JobId) from the initial call to [StartCelebrityDetection](#).

For more information, see [Working with stored videos](#).

[GetCelebrityRecognition](#) returns detected celebrities and the time(s) they are detected in an array ([Celebrities](#)) of [CelebrityRecognition \(p. 265\)](#) objects. Each [CelebrityRecognition](#) contains information about the celebrity in a [CelebrityDetail \(p. 263\)](#) object and the time, [Timestamp](#), the celebrity was detected. This [CelebrityDetail \(p. 263\)](#) object stores information about the detected celebrity's face attributes, a face bounding box, known gender, the celebrity's name, and a confidence estimate.

Note

[GetCelebrityRecognition](#) only returns the default facial attributes (BoundingBox, Confidence, Landmarks, Pose, and Quality). The BoundingBox field only applies to the detected face instance. The other facial attributes listed in the Face object of the following response syntax are not returned. For more information, see [FaceDetail \(p. 302\)](#).

By default, the [Celebrities](#) array is sorted by time (milliseconds from the start of the video). You can also sort the array by celebrity by specifying the value ID in the [SortBy](#) input parameter.

The [CelebrityDetail](#) object includes the celebrity identifier and additional information urls. If you don't store the additional information urls, you can get them later by calling [GetCelebrityInfo \(p. 105\)](#) with the celebrity identifier.

No information is returned for faces not recognized as celebrities.

Use [MaxResults](#) parameter to limit the number of labels returned. If there are more results than specified in [MaxResults](#), the value of [NextToken](#) in the operation response contains a pagination token for getting the next set of results. To get the next page of results, call [GetCelebrityDetection](#) and populate the [NextToken](#) request parameter with the token value returned from the previous call to [GetCelebrityRecognition](#).

Request Syntax

```
{  
  "JobId": "string",  
  "MaxResults": number,  
  "NextToken": "string",  
  "SortBy": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[JobId \(p. 108\)](#)

Job identifier for the required celebrity recognition analysis. You can get the job identifier from a call to `StartCelebrityRecognition`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z0-9-_]+$`

Required: Yes

[MaxResults \(p. 108\)](#)

Maximum number of results to return per paginated call. The largest value you can specify is 1000. If you specify a value greater than 1000, a maximum of 1000 results is returned. The default value is 1000.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

[NextToken \(p. 108\)](#)

If the previous response was incomplete (because there is more recognized celebrities to retrieve), Amazon Rekognition Video returns a pagination token in the response. You can use this pagination token to retrieve the next set of celebrities.

Type: String

Length Constraints: Maximum length of 255.

Required: No

[SortBy \(p. 108\)](#)

Sort to use for celebrities returned in `Celebrities` field. Specify `ID` to sort by the celebrity identifier, specify `TIMESTAMP` to sort by the time the celebrity was recognized.

Type: String

Valid Values: `ID` | `TIMESTAMP`

Required: No

Response Syntax

```
{  
  "Celebrities": [  
    {  
      "Celebrity": {  
        "BoundingBox": {  
          "Height": number,  
          "Left": number,  
          "Top": number,  
          "Width": number  
        },  
      }  
    }  
  ]  
}
```

```

"Confidence": number,
"Face": {
    "AgeRange": {
        "High": number,
        "Low": number
    },
    "Beard": {
        "Confidence": number,
        "Value": boolean
    },
    "BoundingBox": {
        "Height": number,
        "Left": number,
        "Top": number,
        "Width": number
    },
    "Confidence": number,
    "Emotions": [
        {
            "Confidence": number,
            "Type": "string"
        }
    ],
    "Eyeglasses": {
        "Confidence": number,
        "Value": boolean
    },
    "EyesOpen": {
        "Confidence": number,
        "Value": boolean
    },
    "Gender": {
        "Confidence": number,
        "Value": "string"
    },
    "Landmarks": [
        {
            "Type": "string",
            "X": number,
            "Y": number
        }
    ],
    "MouthOpen": {
        "Confidence": number,
        "Value": boolean
    },
    "Mustache": {
        "Confidence": number,
        "Value": boolean
    },
    "Pose": {
        "Pitch": number,
        "Roll": number,
        "Yaw": number
    },
    "Quality": {
        "Brightness": number,
        "Sharpness": number
    },
    "Smile": {
        "Confidence": number,
        "Value": boolean
    },
    "Sunglasses": {
        "Confidence": number,
        "Value": boolean
    }
}

```

```
        },
        "Id": "string",
        "KnownGender": {
            "Type": "string"
        },
        "Name": "string",
        "Urls": [ "string" ]
    },
    "Timestamp": number
}
],
"JobStatus": "string",
"NextToken": "string",
"StatusMessage": "string",
"VideoMetadata": {
    "Codec": "string",
    "ColorRange": "string",
    "DurationMillis": number,
    "Format": "string",
    "FrameHeight": number,
    "FrameRate": number,
    "FrameWidth": number
}
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[Celebrities \(p. 109\)](#)

Array of celebrities recognized in the video.

Type: Array of [CelebrityRecognition \(p. 265\)](#) objects

[JobStatus \(p. 109\)](#)

The current status of the celebrity recognition job.

Type: String

Valid Values: IN_PROGRESS | SUCCEEDED | FAILED

[NextToken \(p. 109\)](#)

If the response is truncated, Amazon Rekognition Video returns this token that you can use in the subsequent request to retrieve the next set of celebrities.

Type: String

Length Constraints: Maximum length of 255.

[StatusMessage \(p. 109\)](#)

If the job fails, StatusMessage provides a descriptive error message.

Type: String

[VideoMetadata \(p. 109\)](#)

Information about a video that Amazon Rekognition Video analyzed. Videometadata is returned in every page of paginated responses from a Amazon Rekognition Video operation.

Type: [VideoMetadata \(p. 387\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)

- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetContentModeration

Gets the inappropriate, unwanted, or offensive content analysis results for a Amazon Rekognition Video analysis started by [StartContentModeration \(p. 201\)](#). For a list of moderation labels in Amazon Rekognition, see [Using the image and video moderation APIs](#).

Amazon Rekognition Video inappropriate or offensive content detection in a stored video is an asynchronous operation. You start analysis by calling [StartContentModeration \(p. 201\)](#) which returns a job identifier (JobId). When analysis finishes, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic registered in the initial call to [StartContentModeration](#). To get the results of the content analysis, first check that the status value published to the Amazon SNS topic is SUCCEEDED. If so, call [GetContentModeration](#) and pass the job identifier (JobId) from the initial call to [StartContentModeration](#).

For more information, see [Working with stored videos](#).

[GetContentModeration](#) returns detected inappropriate, unwanted, or offensive content moderation labels, and the time they are detected, in an array, `ModerationLabels`, of [ContentModerationDetection \(p. 272\)](#) objects.

By default, the moderated labels are returned sorted by time, in milliseconds from the start of the video. You can also sort them by moderated label by specifying NAME for the `SortBy` input parameter.

Since video analysis can return a large number of results, use the `MaxResults` parameter to limit the number of labels returned in a single call to [GetContentModeration](#). If there are more results than specified in `MaxResults`, the value of `NextToken` in the operation response contains a pagination token for getting the next set of results. To get the next page of results, call [GetContentModeration](#) and populate the `NextToken` request parameter with the value of `NextToken` returned from the previous call to [GetContentModeration](#).

For more information, see [Moderating content](#).

Request Syntax

```
{  
  "JobId": "string",  
  "MaxResults": number,  
  "NextToken": "string",  
  "SortBy": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

JobId (p. 114)

The identifier for the inappropriate, unwanted, or offensive content moderation job. Use `JobId` to identify the job in a subsequent call to [GetContentModeration](#).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: Yes

[MaxResults \(p. 114\)](#)

Maximum number of results to return per paginated call. The largest value you can specify is 1000. If you specify a value greater than 1000, a maximum of 1000 results is returned. The default value is 1000.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

[NextToken \(p. 114\)](#)

If the previous response was incomplete (because there is more data to retrieve), Amazon Rekognition returns a pagination token in the response. You can use this pagination token to retrieve the next set of content moderation labels.

Type: String

Length Constraints: Maximum length of 255.

Required: No

[SortBy \(p. 114\)](#)

Sort to use for elements in the `ModerationLabelDetections` array. Use `TIMESTAMP` to sort array elements by the time labels are detected. Use `NAME` to alphabetically group elements for a label together. Within each label group, the array elements are sorted by detection confidence. The default sort is by `TIMESTAMP`.

Type: String

Valid Values: `NAME` | `TIMESTAMP`

Required: No

Response Syntax

```
{  
  "JobStatus": "string",  
  "ModerationLabels": [  
    {  
      "ModerationLabel": {  
        "Confidence": number,  
        "Name": "string",  
        "ParentName": "string"  
      },  
      "Timestamp": number  
    }  
  ],  
  "ModerationModelVersion": "string",  
  "NextToken": "string",  
  "StatusMessage": "string",  
  "VideoMetadata": {  
    "Codec": "string",  
    "ColorRange": "string",  
    "DurationMillis": number,  
    "Format": "string",  
    "Height": number,  
    "Width": number  
  }  
}
```

```
    "FrameHeight": number,  
    "FrameRate": number,  
    "FrameWidth": number  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[JobStatus \(p. 115\)](#)

The current status of the content moderation analysis job.

Type: String

Valid Values: IN_PROGRESS | SUCCEEDED | FAILED

[ModerationLabels \(p. 115\)](#)

The detected inappropriate, unwanted, or offensive content moderation labels and the time(s) they were detected.

Type: Array of [ContentModerationDetection \(p. 272\)](#) objects

[ModerationModelVersion \(p. 115\)](#)

Version number of the moderation detection model that was used to detect inappropriate, unwanted, or offensive content.

Type: String

[NextToken \(p. 115\)](#)

If the response is truncated, Amazon Rekognition Video returns this token that you can use in the subsequent request to retrieve the next set of content moderation labels.

Type: String

Length Constraints: Maximum length of 255.

[StatusMessage \(p. 115\)](#)

If the job fails, StatusMessage provides a descriptive error message.

Type: String

[VideoMetadata \(p. 115\)](#)

Information about a video that Amazon Rekognition analyzed. Videometadata is returned in every page of paginated responses from `GetContentModeration`.

Type: [VideoMetadata \(p. 387\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetFaceDetection

Gets face detection results for a Amazon Rekognition Video analysis started by [StartFaceDetection \(p. 205\)](#).

Face detection with Amazon Rekognition Video is an asynchronous operation. You start face detection by calling [StartFaceDetection \(p. 205\)](#) which returns a job identifier (JobId). When the face detection operation finishes, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic registered in the initial call to [StartFaceDetection](#). To get the results of the face detection operation, first check that the status value published to the Amazon SNS topic is SUCCEEDED. If so, call [GetFaceDetection \(p. 118\)](#) and pass the job identifier (JobId) from the initial call to [StartFaceDetection](#).

`GetFaceDetection` returns an array of detected faces (`Faces`) sorted by the time the faces were detected.

Use `MaxResults` parameter to limit the number of labels returned. If there are more results than specified in `MaxResults`, the value of `NextToken` in the operation response contains a pagination token for getting the next set of results. To get the next page of results, call `GetFaceDetection` and populate the `NextToken` request parameter with the token value returned from the previous call to `GetFaceDetection`.

Request Syntax

```
{  
  "JobId": "string",  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

JobId [\(p. 118\)](#)

Unique identifier for the face detection job. The `JobId` is returned from `StartFaceDetection`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z0-9-_]+$`

Required: Yes

MaxResults [\(p. 118\)](#)

Maximum number of results to return per paginated call. The largest value you can specify is 1000. If you specify a value greater than 1000, a maximum of 1000 results is returned. The default value is 1000.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

NextToken (p. 118)

If the previous response was incomplete (because there are more faces to retrieve), Amazon Rekognition Video returns a pagination token in the response. You can use this pagination token to retrieve the next set of faces.

Type: String

Length Constraints: Maximum length of 255.

Required: No

Response Syntax

```
{  
  "Faces": [  
    {  
      "Face": {  
        "AgeRange": {  
          "High": number,  
          "Low": number  
        },  
        "Beard": {  
          "Confidence": number,  
          "Value": boolean  
        },  
        "BoundingBox": {  
          "Height": number,  
          "Left": number,  
          "Top": number,  
          "Width": number  
        },  
        "Confidence": number,  
        "Emotions": [  
          {  
            "Confidence": number,  
            "Type": "string"  
          }  
        ],  
        "Eyeglasses": {  
          "Confidence": number,  
          "Value": boolean  
        },  
        "EyesOpen": {  
          "Confidence": number,  
          "Value": boolean  
        },  
        "Gender": {  
          "Confidence": number,  
          "Value": "string"  
        },  
        "Landmarks": [  
          {  
            "Type": "string",  
            "X": number,  
            "Y": number  
          }  
        ],  
        "MouthOpen": {  
          "Confidence": number,  
          "Value": boolean  
        }  
      }  
    }  
  ]  
}
```

```

        },
        "Mustache": {
            "Confidence": number,
            "Value": boolean
        },
        "Pose": {
            "Pitch": number,
            "Roll": number,
            "Yaw": number
        },
        "Quality": {
            "Brightness": number,
            "Sharpness": number
        },
        "Smile": {
            "Confidence": number,
            "Value": boolean
        },
        "Sunglasses": {
            "Confidence": number,
            "Value": boolean
        }
    },
    "Timestamp": number
}
],
"JobStatus": "string",
"NextToken": "string",
"StatusMessage": "string",
"VideoMetadata": {
    "Codec": "string",
    "ColorRange": "string",
    "DurationMillis": number,
    "Format": "string",
    "FrameHeight": number,
    "FrameRate": number,
    "FrameWidth": number
}
}

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[Faces \(p. 119\)](#)

An array of faces detected in the video. Each element contains a detected face's details and the time, in milliseconds from the start of the video, the face was detected.

Type: Array of [FaceDetection \(p. 305\)](#) objects

[JobStatus \(p. 119\)](#)

The current status of the face detection job.

Type: String

Valid Values: IN_PROGRESS | SUCCEEDED | FAILED

[NextToken \(p. 119\)](#)

If the response is truncated, Amazon Rekognition returns this token that you can use in the subsequent request to retrieve the next set of faces.

Type: String

Length Constraints: Maximum length of 255.

[StatusMessage \(p. 119\)](#)

If the job fails, StatusMessage provides a descriptive error message.

Type: String

[VideoMetadata \(p. 119\)](#)

Information about a video that Amazon Rekognition Video analyzed. Videodata is returned in every page of paginated responses from a Amazon Rekognition video operation.

Type: [VideoMetadata \(p. 387\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetFaceSearch

Gets the face search results for Amazon Rekognition Video face search started by [StartFaceSearch \(p. 209\)](#). The search returns faces in a collection that match the faces of persons detected in a video. It also includes the time(s) that faces are matched in the video.

Face search in a video is an asynchronous operation. You start face search by calling to [StartFaceSearch \(p. 209\)](#) which returns a job identifier (JobId). When the search operation finishes, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic registered in the initial call to StartFaceSearch. To get the search results, first check that the status value published to the Amazon SNS topic is SUCCEEDED. If so, call GetFaceSearch and pass the job identifier (JobId) from the initial call to StartFaceSearch.

For more information, see [Searching faces in a collection](#).

The search results are returned in an array, Persons, of [PersonMatch \(p. 338\)](#) objects. EachPersonMatch element contains details about the matching faces in the input collection, person information (facial attributes, bounding boxes, and person identifier) for the matched person, and the time the person was matched in the video.

Note

GetFaceSearch only returns the default facial attributes (BoundingBox, Confidence, Landmarks, Pose, and Quality). The other facial attributes listed in the Face object of the following response syntax are not returned. For more information, see [FaceDetail \(p. 302\)](#).

By default, the Persons array is sorted by the time, in milliseconds from the start of the video, persons are matched. You can also sort by persons by specifying INDEX for the SORTBY input parameter.

Request Syntax

```
{  
  "JobId": "string",  
  "MaxResults": number,  
  "NextToken": "string",  
  "SortBy": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

JobId (p. 123)

The job identifier for the search request. You get the job identifier from an initial call to StartFaceSearch.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: Yes

[MaxResults \(p. 123\)](#)

Maximum number of results to return per paginated call. The largest value you can specify is 1000. If you specify a value greater than 1000, a maximum of 1000 results is returned. The default value is 1000.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

[NextToken \(p. 123\)](#)

If the previous response was incomplete (because there is more search results to retrieve), Amazon Rekognition Video returns a pagination token in the response. You can use this pagination token to retrieve the next set of search results.

Type: String

Length Constraints: Maximum length of 255.

Required: No

[SortBy \(p. 123\)](#)

Sort to use for grouping faces in the response. Use TIMESTAMP to group faces by the time that they are recognized. Use INDEX to sort by recognized faces.

Type: String

Valid Values: INDEX | TIMESTAMP

Required: No

Response Syntax

```
{  
  "JobStatus": "string",  
  "NextToken": "string",  
  "Persons": [  
    {  
      "FaceMatches": [  
        {  
          "Face": {  
            "BoundingBox": {  
              "Height": number,  
              "Left": number,  
              "Top": number,  
              "Width": number  
            },  
            "Confidence": number,  
            "ExternalImageId": "string",  
            "FaceId": "string",  
            "ImageId": "string",  
            "IndexFacesModelVersion": "string"  
          },  
          "Similarity": number  
        }  
      ],  
      "Person": {  
        "BoundingBox": {  
          "Height": number,  
          "Width": number  
        },  
        "FaceMatches": [  
          {  
            "Face": {  
              "BoundingBox": {  
                "Height": number,  
                "Left": number,  
                "Top": number,  
                "Width": number  
              },  
              "Confidence": number,  
              "ExternalImageId": "string",  
              "FaceId": "string",  
              "ImageId": "string",  
              "IndexFacesModelVersion": "string"  
            },  
            "Similarity": number  
          }  
        ]  
      }  
    }  
  ]  
}
```

```

    "Left": number,
    "Top": number,
    "Width": number
},
"Face": {
    "AgeRange": {
        "High": number,
        "Low": number
    },
    "Beard": {
        "Confidence": number,
        "Value": boolean
    },
    "BoundingBox": {
        "Height": number,
        "Left": number,
        "Top": number,
        "Width": number
    },
    "Confidence": number,
    "Emotions": [
        {
            "Confidence": number,
            "Type": "string"
        }
    ],
    "Eyeglasses": {
        "Confidence": number,
        "Value": boolean
    },
    "EyesOpen": {
        "Confidence": number,
        "Value": boolean
    },
    "Gender": {
        "Confidence": number,
        "Value": "string"
    },
    "Landmarks": [
        {
            "Type": "string",
            "X": number,
            "Y": number
        }
    ],
    "MouthOpen": {
        "Confidence": number,
        "Value": boolean
    },
    "Mustache": {
        "Confidence": number,
        "Value": boolean
    },
    "Pose": {
        "Pitch": number,
        "Roll": number,
        "Yaw": number
    },
    "Quality": {
        "Brightness": number,
        "Sharpness": number
    },
    "Smile": {
        "Confidence": number,
        "Value": boolean
    }
},

```

```

        "Sunglasses": {
            "Confidence": number,
            "Value": boolean
        },
        "Index": number
    },
    "Timestamp": number
}
],
"StatusMessage": "string",
"VideoMetadata": {
    "Codec": "string",
    "ColorRange": "string",
    "DurationMillis": number,
    "Format": "string",
    "FrameHeight": number,
    "FrameRate": number,
    "FrameWidth": number
}
}
}

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[JobStatus \(p. 124\)](#)

The current status of the face search job.

Type: String

Valid Values: IN_PROGRESS | SUCCEEDED | FAILED

[NextToken \(p. 124\)](#)

If the response is truncated, Amazon Rekognition Video returns this token that you can use in the subsequent request to retrieve the next set of search results.

Type: String

Length Constraints: Maximum length of 255.

[Persons \(p. 124\)](#)

An array of persons, [PersonMatch \(p. 338\)](#), in the video whose face(s) match the face(s) in an Amazon Rekognition collection. It also includes time information for when persons are matched in the video. You specify the input collection in an initial call to `StartFaceSearch`. Each `Persons` element includes a time the person was matched, face match details (`FaceMatches`) for matching faces in the collection, and person information (`Person`) for the matched person.

Type: Array of [PersonMatch \(p. 338\)](#) objects

[StatusMessage \(p. 124\)](#)

If the job fails, `StatusMessage` provides a descriptive error message.

Type: String

[VideoMetadata \(p. 124\)](#)

Information about a video that Amazon Rekognition analyzed. `Videometadata` is returned in every page of paginated responses from a Amazon Rekognition Video operation.

Type: [VideoMetadata \(p. 387\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)

- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetLabelDetection

Gets the label detection results of a Amazon Rekognition Video analysis started by [StartLabelDetection \(p. 213\)](#).

The label detection operation is started by a call to [StartLabelDetection \(p. 213\)](#) which returns a job identifier (JobId). When the label detection operation finishes, Amazon Rekognition publishes a completion status to the Amazon Simple Notification Service topic registered in the initial call to StartLabelDetection. To get the results of the label detection operation, first check that the status value published to the Amazon SNS topic is SUCCEEDED. If so, call [GetLabelDetection \(p. 129\)](#) and pass the job identifier (JobId) from the initial call to StartLabelDetection.

GetLabelDetection returns an array of detected labels (Labels) sorted by the time the labels were detected. You can also sort by the label name by specifying NAME for the SortBy input parameter.

The labels returned include the label name, the percentage confidence in the accuracy of the detected label, and the time the label was detected in the video.

The returned labels also include bounding box information for common objects, a hierarchical taxonomy of detected labels, and the version of the label model used for detection.

Use MaxResults parameter to limit the number of labels returned. If there are more results than specified in MaxResults, the value of NextToken in the operation response contains a pagination token for getting the next set of results. To get the next page of results, call GetLabelDetection and populate the NextToken request parameter with the token value returned from the previous call to GetLabelDetection.

Request Syntax

```
{  
  "JobId": "string",  
  "MaxResults": number,  
  "NextToken": "string",  
  "SortBy": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

JobId (p. 129)

Job identifier for the label detection operation for which you want results returned. You get the job identifier from an initial call to StartLabelDetection.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: Yes

[MaxResults \(p. 129\)](#)

Maximum number of results to return per paginated call. The largest value you can specify is 1000. If you specify a value greater than 1000, a maximum of 1000 results is returned. The default value is 1000.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

[NextToken \(p. 129\)](#)

If the previous response was incomplete (because there are more labels to retrieve), Amazon Rekognition Video returns a pagination token in the response. You can use this pagination token to retrieve the next set of labels.

Type: String

Length Constraints: Maximum length of 255.

Required: No

[SortBy \(p. 129\)](#)

Sort to use for elements in the Labels array. Use TIMESTAMP to sort array elements by the time labels are detected. Use NAME to alphabetically group elements for a label together. Within each label group, the array elements are sorted by detection confidence. The default sort is by TIMESTAMP.

Type: String

Valid Values: NAME | TIMESTAMP

Required: No

Response Syntax

```
{  
  "JobStatus": "string",  
  "LabelModelVersion": "string",  
  "Labels": [  
    {  
      "Label": {  
        "Aliases": [  
          {  
            "Name": "string"  
          }  
        ],  
        "Categories": [  
          {  
            "Name": "string"  
          }  
        ],  
        "Confidence": number,  
        "Instances": [  
          {  
            "BoundingBox": {  
              "Height": number,  
              "Left": number,  
              "Top": number,  
              "Width": number  
            }  
          }  
        ]  
      }  
    }  
  ]  
}
```

```

        "Top": number,
        "Width": number
    },
    "Confidence": number,
    "DominantColors": [
        {
            "Blue": number,
            "CSSColor": "string",
            "Green": number,
            "HexCode": "string",
            "PixelPercent": number,
            "Red": number,
            "SimplifiedColor": "string"
        }
    ]
},
"Name": "string",
"Parents": [
    {
        "Name": "string"
    }
],
"Timestamp": number
},
],
"NextToken": "string",
"StatusMessage": "string",
"VideoMetadata": {
    "Codec": "string",
    "ColorRange": "string",
    "DurationMillis": number,
    "Format": "string",
    "FrameHeight": number,
    "FrameRate": number,
    "FrameWidth": number
}
}
}

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

JobStatus (p. 130)

The current status of the label detection job.

Type: String

Valid Values: IN_PROGRESS | SUCCEEDED | FAILED

LabelModelVersion (p. 130)

Version number of the label detection model that was used to detect labels.

Type: String

Labels (p. 130)

An array of labels detected in the video. Each element contains the detected label and the time, in milliseconds from the start of the video, that the label was detected.

Type: Array of [LabelDetection \(p. 328\)](#) objects

NextToken (p. 130)

If the response is truncated, Amazon Rekognition Video returns this token that you can use in the subsequent request to retrieve the next set of labels.

Type: String

Length Constraints: Maximum length of 255.

StatusMessage (p. 130)

If the job fails, StatusMessage provides a descriptive error message.

Type: String

VideoMetadata (p. 130)

Information about a video that Amazon Rekognition Video analyzed. Videometadata is returned in every page of paginated responses from a Amazon Rekognition video operation.

Type: [VideoMetadata \(p. 387\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetPersonTracking

Gets the path tracking results of a Amazon Rekognition Video analysis started by [StartPersonTracking \(p. 217\)](#).

The person path tracking operation is started by a call to `StartPersonTracking` which returns a job identifier (`JobId`). When the operation finishes, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic registered in the initial call to `StartPersonTracking`.

To get the results of the person path tracking operation, first check that the status value published to the Amazon SNS topic is `SUCCEEDED`. If so, call [GetPersonTracking \(p. 134\)](#) and pass the job identifier (`JobId`) from the initial call to `StartPersonTracking`.

`GetPersonTracking` returns an array, `Persons`, of tracked persons and the time(s) their paths were tracked in the video.

Note

`GetPersonTracking` only returns the default facial attributes (`BoundingBox`, `Confidence`, `Landmarks`, `Pose`, and `Quality`). The other facial attributes listed in the `Face` object of the following response syntax are not returned.

For more information, see [FaceDetail \(p. 302\)](#).

By default, the array is sorted by the time(s) a person's path is tracked in the video. You can sort by tracked persons by specifying `INDEX` for the `SortBy` input parameter.

Use the `MaxResults` parameter to limit the number of items returned. If there are more results than specified in `MaxResults`, the value of `NextToken` in the operation response contains a pagination token for getting the next set of results. To get the next page of results, call `GetPersonTracking` and populate the `NextToken` request parameter with the token value returned from the previous call to `GetPersonTracking`.

Request Syntax

```
{  
  "JobId": "string",  
  "MaxResults": number,  
  "NextToken": "string",  
  "SortBy": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

JobId (p. 134)

The identifier for a job that tracks persons in a video. You get the `JobId` from a call to `StartPersonTracking`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: Yes

[MaxResults \(p. 134\)](#)

Maximum number of results to return per paginated call. The largest value you can specify is 1000. If you specify a value greater than 1000, a maximum of 1000 results is returned. The default value is 1000.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

[NextToken \(p. 134\)](#)

If the previous response was incomplete (because there are more persons to retrieve), Amazon Rekognition Video returns a pagination token in the response. You can use this pagination token to retrieve the next set of persons.

Type: String

Length Constraints: Maximum length of 255.

Required: No

[SortBy \(p. 134\)](#)

Sort to use for elements in the Persons array. Use TIMESTAMP to sort array elements by the time persons are detected. Use INDEX to sort by the tracked persons. If you sort by INDEX, the array elements for each person are sorted by detection confidence. The default sort is by TIMESTAMP.

Type: String

Valid Values: INDEX | TIMESTAMP

Required: No

Response Syntax

```
{  
  "JobStatus": "string",  
  "NextToken": "string",  
  "Persons": [  
    {  
      "Person": {  
        "BoundingBox": {  
          "Height": number,  
          "Left": number,  
          "Top": number,  
          "Width": number  
        },  
        "Face": {  
          "AgeRange": {  
            "High": number,  
            "Low": number  
          },  
          "Beard": {  
            "Confidence": number,  
            "Value": boolean  
          }  
        }  
      }  
    }  
  ]  
}
```

```

        },
        "BoundingBox": {
            "Height": number,
            "Left": number,
            "Top": number,
            "Width": number
        },
        "Confidence": number,
        "Emotions": [
            {
                "Confidence": number,
                "Type": "string"
            }
        ],
        "Eyeglasses": {
            "Confidence": number,
            "Value": boolean
        },
        "EyesOpen": {
            "Confidence": number,
            "Value": boolean
        },
        "Gender": {
            "Confidence": number,
            "Value": "string"
        },
        "Landmarks": [
            {
                "Type": "string",
                "X": number,
                "Y": number
            }
        ],
        "MouthOpen": {
            "Confidence": number,
            "Value": boolean
        },
        "Mustache": {
            "Confidence": number,
            "Value": boolean
        },
        "Pose": {
            "Pitch": number,
            "Roll": number,
            "Yaw": number
        },
        "Quality": {
            "Brightness": number,
            "Sharpness": number
        },
        "Smile": {
            "Confidence": number,
            "Value": boolean
        },
        "Sunglasses": {
            "Confidence": number,
            "Value": boolean
        }
    },
    "Index": number
},
"Timestamp": number
}
],
"StatusMessage": "string",
"VideoMetadata": {

```

```
  "Codec": "string",
  "ColorRange": "string",
  "DurationMillis": number,
  "Format": "string",
  "FrameHeight": number,
  "FrameRate": number,
  "FrameWidth": number
}
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[JobStatus \(p. 135\)](#)

The current status of the person tracking job.

Type: String

Valid Values: IN_PROGRESS | SUCCEEDED | FAILED

[NextToken \(p. 135\)](#)

If the response is truncated, Amazon Rekognition Video returns this token that you can use in the subsequent request to retrieve the next set of persons.

Type: String

Length Constraints: Maximum length of 255.

[Persons \(p. 135\)](#)

An array of the persons detected in the video and the time(s) their path was tracked throughout the video. An array element will exist for each time a person's path is tracked.

Type: Array of [PersonDetection \(p. 337\)](#) objects

[StatusMessage \(p. 135\)](#)

If the job fails, StatusMessage provides a descriptive error message.

Type: String

[VideoMetadata \(p. 135\)](#)

Information about a video that Amazon Rekognition Video analyzed. Videometadata is returned in every page of paginated responses from a Amazon Rekognition Video operation.

Type: [VideoMetadata \(p. 387\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetSegmentDetection

Gets the segment detection results of a Amazon Rekognition Video analysis started by [StartSegmentDetection \(p. 224\)](#).

Segment detection with Amazon Rekognition Video is an asynchronous operation. You start segment detection by calling [StartSegmentDetection \(p. 224\)](#) which returns a job identifier (JobId). When the segment detection operation finishes, Amazon Rekognition publishes a completion status to the Amazon Simple Notification Service topic registered in the initial call to [StartSegmentDetection](#). To get the results of the segment detection operation, first check that the status value published to the Amazon SNS topic is SUCCEEDED. If so, call [GetSegmentDetection](#) and pass the job identifier (JobId) from the initial call of [StartSegmentDetection](#).

[GetSegmentDetection](#) returns detected segments in an array (Segments) of [SegmentDetection \(p. 355\)](#) objects. Segments is sorted by the segment types specified in the SegmentTypes input parameter of [StartSegmentDetection](#). Each element of the array includes the detected segment, the percentage confidence in the accuracy of the detected segment, the type of the segment, and the frame in which the segment was detected.

Use SelectedSegmentTypes to find out the type of segment detection requested in the call to [StartSegmentDetection](#).

Use the MaxResults parameter to limit the number of segment detections returned. If there are more results than specified in MaxResults, the value of NextToken in the operation response contains a pagination token for getting the next set of results. To get the next page of results, call [GetSegmentDetection](#) and populate the NextToken request parameter with the token value returned from the previous call to [GetSegmentDetection](#).

For more information, see [Detecting video segments in stored video](#).

Request Syntax

```
{  
  "JobId": "string",  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[JobId \(p. 139\)](#)

Job identifier for the text detection operation for which you want results returned. You get the job identifier from an initial call to [StartSegmentDetection](#).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: Yes

[MaxResults \(p. 139\)](#)

Maximum number of results to return per paginated call. The largest value you can specify is 1000.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

[NextToken \(p. 139\)](#)

If the response is truncated, Amazon Rekognition Video returns this token that you can use in the subsequent request to retrieve the next set of text.

Type: String

Length Constraints: Maximum length of 255.

Required: No

Response Syntax

```
{  
  "AudioMetadata": [  
    {  
      "Codec": "string",  
      "DurationMillis": number,  
      "NumberOfChannels": number,  
      "SampleRate": number  
    }  
  ],  
  "JobStatus": "string",  
  "NextToken": "string",  
  "Segments": [  
    {  
      "DurationFrames": number,  
      "DurationMillis": number,  
      "DurationSMPTE": "string",  
      "EndFrameNumber": number,  
      "EndTimecodeSMPTE": "string",  
      "EndTimestampMillis": number,  
      "ShotSegment": {  
        "Confidence": number,  
        "Index": number  
      },  
      "StartFrameNumber": number,  
      "StartTimecodeSMPTE": "string",  
      "StartTimestampMillis": number,  
      "TechnicalCueSegment": {  
        "Confidence": number,  
        "Type": "string"  
      },  
      "Type": "string"  
    }  
  ],  
  "SelectedSegmentTypes": [  
    {  
      "ModelVersion": "string",  
      "Type": "string"  
    }  
  ],  
  "StatusMessage": "string",  
}
```

```
"VideoMetadata": [  
  {  
    "Codec": "string",  
    "ColorRange": "string",  
    "DurationMillis": number,  
    "Format": "string",  
    "FrameHeight": number,  
    "FrameRate": number,  
    "FrameWidth": number  
  }  
]
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[AudioMetadata \(p. 140\)](#)

An array of [AudioMetadata \(p. 256\)](#) objects. There can be multiple audio streams. Each AudioMetadata object contains metadata for a single audio stream. Audio information in an AudioMetadata objects includes the audio codec, the number of audio channels, the duration of the audio stream, and the sample rate. Audio metadata is returned in each page of information returned by `GetSegmentDetection`.

Type: Array of [AudioMetadata \(p. 256\)](#) objects

[JobStatus \(p. 140\)](#)

Current status of the segment detection job.

Type: String

Valid Values: `IN_PROGRESS` | `SUCCEEDED` | `FAILED`

[NextToken \(p. 140\)](#)

If the previous response was incomplete (because there are more labels to retrieve), Amazon Rekognition Video returns a pagination token in the response. You can use this pagination token to retrieve the next set of text.

Type: String

Length Constraints: Maximum length of 255.

[Segments \(p. 140\)](#)

An array of segments detected in a video. The array is sorted by the segment types (`TECHNICAL_CUE` or `SHOT`) specified in the `SegmentTypes` input parameter of `StartSegmentDetection`. Within each segment type the array is sorted by timestamp values.

Type: Array of [SegmentDetection \(p. 355\)](#) objects

[SelectedSegmentTypes \(p. 140\)](#)

An array containing the segment types requested in the call to `StartSegmentDetection`.

Type: Array of [SegmentTypeInfo \(p. 358\)](#) objects

[StatusMessage \(p. 140\)](#)

If the job fails, `StatusMessage` provides a descriptive error message.

Type: String

VideoMetadata (p. 140)

Currently, Amazon Rekognition Video returns a single [VideoMetadata \(p. 387\)](#) object in the VideoMetadata array. The object contains information about the video stream in the input file that Amazon Rekognition Video chose to analyze. The VideoMetadata object includes the video codec, video format and other information. Video metadata is returned in each page of information returned by GetSegmentDetection.

Type: Array of [VideoMetadata \(p. 387\)](#) objects

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetTextDetection

Gets the text detection results of a Amazon Rekognition Video analysis started by [StartTextDetection \(p. 231\)](#).

Text detection with Amazon Rekognition Video is an asynchronous operation. You start text detection by calling [StartTextDetection \(p. 231\)](#) which returns a job identifier (JobId). When the text detection operation finishes, Amazon Rekognition publishes a completion status to the Amazon Simple Notification Service topic registered in the initial call to [StartTextDetection](#). To get the results of the text detection operation, first check that the status value published to the Amazon SNS topic is **SUCCEEDED**. If so, call [GetTextDetection](#) and pass the job identifier (JobId) from the initial call of [StartTextDetection](#).

[GetTextDetection](#) returns an array of detected text ([TextDetections](#)) sorted by the time the text was detected, up to 50 words per frame of video.

Each element of the array includes the detected text, the percentage confidence in the accuracy of the detected text, the time the text was detected, bounding box information for where the text was located, and unique identifiers for words and their lines.

Use [MaxResults](#) parameter to limit the number of text detections returned. If there are more results than specified in [MaxResults](#), the value of [NextToken](#) in the operation response contains a pagination token for getting the next set of results. To get the next page of results, call [GetTextDetection](#) and populate the [NextToken](#) request parameter with the token value returned from the previous call to [GetTextDetection](#).

Request Syntax

```
{  
  "JobId": "string",  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

JobId (p. 144)

Job identifier for the text detection operation for which you want results returned. You get the job identifier from an initial call to [StartTextDetection](#).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: Yes

MaxResults (p. 144)

Maximum number of results to return per paginated call. The largest value you can specify is 1000.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

[NextToken \(p. 144\)](#)

If the previous response was incomplete (because there are more labels to retrieve), Amazon Rekognition Video returns a pagination token in the response. You can use this pagination token to retrieve the next set of text.

Type: String

Length Constraints: Maximum length of 255.

Required: No

Response Syntax

```
{  
    "JobStatus": "string",  
    "NextToken": "string",  
    "StatusMessage": "string",  
    "TextDetections": [  
        {  
            "TextDetection": {  
                "Confidence": number,  
                "DetectedText": "string",  
                "Geometry": {  
                    "BoundingBox": {  
                        "Height": number,  
                        "Left": number,  
                        "Top": number,  
                        "Width": number  
                    },  
                    "Polygon": [  
                        {  
                            "X": number,  
                            "Y": number  
                        }  
                    ]  
                },  
                "Id": number,  
                "ParentId": number,  
                "Type": "string"  
            },  
            "Timestamp": number  
        }  
    ],  
    "TextModelVersion": "string",  
    "VideoMetadata": {  
        "Codec": "string",  
        "ColorRange": "string",  
        "DurationMillis": number,  
        "Format": "string",  
        "FrameHeight": number,  
        "FrameRate": number,  
        "FrameWidth": number  
    }  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

JobStatus (p. 145)

Current status of the text detection job.

Type: String

Valid Values: IN_PROGRESS | SUCCEEDED | FAILED

NextToken (p. 145)

If the response is truncated, Amazon Rekognition Video returns this token that you can use in the subsequent request to retrieve the next set of text.

Type: String

Length Constraints: Maximum length of 255.

StatusMessage (p. 145)

If the job fails, StatusMessage provides a descriptive error message.

Type: String

TextDetections (p. 145)

An array of text detected in the video. Each element contains the detected text, the time in milliseconds from the start of the video that the text was detected, and where it was detected on the screen.

Type: Array of [TextDetectionResult \(p. 381\)](#) objects

TextModelVersion (p. 145)

Version number of the text detection model that was used to detect text.

Type: String

VideoMetadata (p. 145)

Information about a video that Amazon Rekognition analyzed. Videometadata is returned in every page of paginated responses from a Amazon Rekognition video operation.

Type: [VideoMetadata \(p. 387\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

IndexFaces

Detects faces in the input image and adds them to the specified collection.

Amazon Rekognition doesn't save the actual faces that are detected. Instead, the underlying detection algorithm first detects the faces in the input image. For each face, the algorithm extracts facial features into a feature vector, and stores it in the backend database. Amazon Rekognition uses feature vectors when it performs face match and search operations using the [SearchFaces \(p. 188\)](#) and [SearchFacesByImage \(p. 192\)](#) operations.

For more information, see [Adding faces to a collection](#).

To get the number of faces in a collection, call [DescribeCollection \(p. 55\)](#).

If you're using version 1.0 of the face detection model, IndexFaces indexes the 15 largest faces in the input image. Later versions of the face detection model index the 100 largest faces in the input image.

If you're using version 4 or later of the face model, image orientation information is not returned in the `OrientationCorrection` field.

To determine which version of the model you're using, call [DescribeCollection \(p. 55\)](#) and supply the collection ID. You can also get the model version from the value of `FaceModelVersion` in the response from IndexFaces

For more information, see [Model versioning](#).

If you provide the optional `ExternalImageId` for the input image you provided, Amazon Rekognition associates this ID with all faces that it detects. When you call the [ListFaces \(p. 167\)](#) operation, the response returns the external ID. You can use this external image ID to create a client-side index to associate the faces with each image. You can then use the index to find all faces in an image.

You can specify the maximum number of faces to index with the `MaxFaces` input parameter. This is useful when you want to index the largest faces in an image and don't want to index smaller faces, such as those belonging to people standing in the background.

The `QualityFilter` input parameter allows you to filter out detected faces that don't meet a required quality bar. The quality bar is based on a variety of common use cases. By default, IndexFaces chooses the quality bar that's used to filter faces. You can also explicitly choose the quality bar. Use `QualityFilter`, to set the quality bar by specifying `LOW`, `MEDIUM`, or `HIGH`. If you do not want to filter detected faces, specify `NONE`.

Note

To use quality filtering, you need a collection associated with version 3 of the face model or higher. To get the version of the face model associated with a collection, call [DescribeCollection \(p. 55\)](#).

Information about faces detected in an image, but not indexed, is returned in an array of [UnindexedFace \(p. 384\)](#) objects, `UnindexedFaces`. Faces aren't indexed for reasons such as:

- The number of faces detected exceeds the value of the `MaxFaces` request parameter.
- The face is too small compared to the image dimensions.
- The face is too blurry.
- The image is too dark.
- The face has an extreme pose.
- The face doesn't have enough detail to be suitable for face search.

In response, the IndexFaces operation returns an array of metadata for all detected faces, `FaceRecords`. This includes:

- The bounding box, `BoundingBox`, of the detected face.
- A confidence value, `Confidence`, which indicates the confidence that the bounding box contains a face.
- A face ID, `FaceId`, assigned by the service for each face that's detected and stored.
- An image ID, `ImageId`, assigned by the service for the input image.

If you request all facial attributes (by using the `detectionAttributes` parameter), Amazon Rekognition returns detailed facial attributes, such as facial landmarks (for example, location of eye and mouth) and other facial attributes. If you provide the same image, specify the same collection, and use the same external ID in the `IndexFaces` operation, Amazon Rekognition doesn't save duplicate face metadata.

The input image is passed either as base64-encoded image bytes, or as a reference to an image in an Amazon S3 bucket. If you use the AWS CLI to call Amazon Rekognition operations, passing image bytes isn't supported. The image must be formatted as a PNG or JPEG file.

This operation requires permissions to perform the `rekognition:IndexFaces` action.

Request Syntax

```
{
  "CollectionId": "string",
  "DetectionAttributes": [ "string" ],
  "ExternalImageId": "string",
  "Image": {
    "Bytes": blob,
    "S3Object": {
      "Bucket": "string",
      "Name": "string",
      "Version": "string"
    }
  },
  "MaxFaces": number,
  "QualityFilter": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

CollectionId (p. 149)

The ID of an existing collection to which you want to add the faces that are detected in the input images.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\+]+

Required: Yes

[DetectionAttributes \(p. 149\)](#)

An array of facial attributes that you want to be returned. This can be the default list of attributes or all attributes. If you don't specify a value for `Attributes` or if you specify `["DEFAULT"]`, the API returns the following subset of facial attributes: `BoundingBox`, `Confidence`, `Pose`, `Quality`, and `Landmarks`. If you provide `["ALL"]`, all facial attributes are returned, but the operation takes longer to complete.

If you provide both, `["ALL"]`, `["DEFAULT"]`, the service uses a logical AND operator to determine which attributes to return (in this case, all attributes).

Type: Array of strings

Valid Values: `DEFAULT` | `ALL`

Required: No

[ExternalImageId \(p. 149\)](#)

The ID you want to assign to all the faces detected in the image.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_.\-\:]+`

Required: No

[Image \(p. 149\)](#)

The input image as base64-encoded bytes or an S3 object. If you use the AWS CLI to call Amazon Rekognition operations, passing base64-encoded image bytes isn't supported.

If you are using an AWS SDK to call Amazon Rekognition, you might not need to base64-encode image bytes passed using the `Bytes` field. For more information, see [Image specifications](#).

Type: [Image \(p. 317\)](#) object

Required: Yes

[MaxFaces \(p. 149\)](#)

The maximum number of faces to index. The value of `MaxFaces` must be greater than or equal to 1. `IndexFaces` returns no more than 100 detected faces in an image, even if you specify a larger value for `MaxFaces`.

If `IndexFaces` detects more faces than the value of `MaxFaces`, the faces with the lowest quality are filtered out first. If there are still more faces than the value of `MaxFaces`, the faces with the smallest bounding boxes are filtered out (up to the number that's needed to satisfy the value of `MaxFaces`). Information about the unindexed faces is available in the `UnindexedFaces` array.

The faces that are returned by `IndexFaces` are sorted by the largest face bounding box size to the smallest size, in descending order.

`MaxFaces` can be used with a collection associated with any version of the face model.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

QualityFilter (p. 149)

A filter that specifies a quality bar for how much filtering is done to identify faces. Filtered faces aren't indexed. If you specify AUTO, Amazon Rekognition chooses the quality bar. If you specify LOW, MEDIUM, or HIGH, filtering removes all faces that don't meet the chosen quality bar. The default value is AUTO. The quality bar is based on a variety of common use cases. Low-quality detections can occur for a number of reasons. Some examples are an object that's misidentified as a face, a face that's too blurry, or a face with a pose that's too extreme to use. If you specify NONE, no filtering is performed.

To use quality filtering, the collection you are using must be associated with version 3 of the face model or higher.

Type: String

Valid Values: NONE | AUTO | LOW | MEDIUM | HIGH

Required: No

Response Syntax

```
{
  "FaceModelVersion": "string",
  "FaceRecords": [
    {
      "Face": {
        "BoundingBox": {
          "Height": number,
          "Left": number,
          "Top": number,
          "Width": number
        },
        "Confidence": number,
        "ExternalImageId": "string",
        "FaceId": "string",
        "ImageId": "string",
        "IndexFacesModelVersion": "string"
      },
      "FaceDetail": {
        "AgeRange": {
          "High": number,
          "Low": number
        },
        "Beard": {
          "Confidence": number,
          "Value": boolean
        },
        "BoundingBox": {
          "Height": number,
          "Left": number,
          "Top": number,
          "Width": number
        },
        "Confidence": number,
        "Emotions": [
          {
            "Confidence": number,
            "Type": "string"
          }
        ],
        "Eyeglasses": {
          "Confidence": number,
        }
      }
    }
  ]
}
```

```

        "Value": boolean
    },
    "EyesOpen": {
        "Confidence": number,
        "Value": boolean
    },
    "Gender": {
        "Confidence": number,
        "Value": "string"
    },
    "Landmarks": [
        {
            "Type": "string",
            "X": number,
            "Y": number
        }
    ],
    "MouthOpen": {
        "Confidence": number,
        "Value": boolean
    },
    "Mustache": {
        "Confidence": number,
        "Value": boolean
    },
    "Pose": {
        "Pitch": number,
        "Roll": number,
        "Yaw": number
    },
    "Quality": {
        "Brightness": number,
        "Sharpness": number
    },
    "Smile": {
        "Confidence": number,
        "Value": boolean
    },
    "Sunglasses": {
        "Confidence": number,
        "Value": boolean
    }
}
],
"OrientationCorrection": "string",
"UnindexedFaces": [
    {
        "FaceDetail": {
            "AgeRange": {
                "High": number,
                "Low": number
            },
            "Beard": {
                "Confidence": number,
                "Value": boolean
            },
            "BoundingBox": {
                "Height": number,
                "Left": number,
                "Top": number,
                "Width": number
            },
            "Confidence": number,
            "Emotions": [
                {

```

```

        "Confidence": number,
        "Type": "string"
    }
],
"Eyeglasses": {
    "Confidence": number,
    "Value": boolean
},
"EyesOpen": {
    "Confidence": number,
    "Value": boolean
},
"Gender": {
    "Confidence": number,
    "Value": "string"
},
"Landmarks": [
    {
        "Type": "string",
        "X": number,
        "Y": number
    }
],
"MouthOpen": {
    "Confidence": number,
    "Value": boolean
},
"Mustache": {
    "Confidence": number,
    "Value": boolean
},
"Pose": {
    "Pitch": number,
    "Roll": number,
    "Yaw": number
},
"Quality": {
    "Brightness": number,
    "Sharpness": number
},
"Smile": {
    "Confidence": number,
    "Value": boolean
},
"Sunglasses": {
    "Confidence": number,
    "Value": boolean
}
],
"Reasons": [ "string" ]
}
]
}

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[FaceModelVersion \(p. 151\)](#)

The version number of the face detection model that's associated with the input collection (CollectionId).

Type: String

[FaceRecords \(p. 151\)](#)

An array of faces detected and added to the collection. For more information, see [Searching faces in a collection](#).

Type: Array of [FaceRecord \(p. 307\)](#) objects

[OrientationCorrection \(p. 151\)](#)

If your collection is associated with a face detection model that's later than version 3.0, the value of `OrientationCorrection` is always null and no orientation information is returned.

If your collection is associated with a face detection model that's version 3.0 or earlier, the following applies:

- If the input image is in .jpeg format, it might contain exchangeable image file format (Exif) metadata that includes the image's orientation. Amazon Rekognition uses this orientation information to perform image correction - the bounding box coordinates are translated to represent object locations after the orientation information in the Exif metadata is used to correct the image orientation. Images in .png format don't contain Exif metadata. The value of `OrientationCorrection` is null.
- If the image doesn't contain orientation information in its Exif metadata, Amazon Rekognition returns an estimated orientation (ROTATE_0, ROTATE_90, ROTATE_180, ROTATE_270). Amazon Rekognition doesn't perform image correction for images. The bounding box coordinates aren't translated and represent the object locations before the image is rotated.

Bounding box information is returned in the `FaceRecords` array. You can get the version of the face detection model by calling [DescribeCollection \(p. 55\)](#).

Type: String

Valid Values: ROTATE_0 | ROTATE_90 | ROTATE_180 | ROTATE_270

[UnindexedFaces \(p. 151\)](#)

An array of faces that were detected in the image but weren't indexed. They weren't indexed because the quality filter identified them as low quality, or the `MaxFaces` request parameter filtered them out. To use the quality filter, you specify the `QualityFilter` request parameter.

Type: Array of [UnindexedFace \(p. 384\)](#) objects

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

ImageTooLargeException

The input image size exceeds the allowed limit. If you are calling [DetectProtectiveEquipment \(p. 95\)](#), the image size or resolution exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidImageFormatException

The provided image format is not supported.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ServiceQuotaExceededException

The size of the resource exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListCollections

Returns list of collection IDs in your account. If the result is truncated, the response also provides a `NextToken` that you can use in the subsequent request to fetch the next set of collection IDs.

For an example, see [Listing collections](#).

This operation requires permissions to perform the `rekognition:ListCollections` action.

Request Syntax

```
{  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

MaxResults (p. 157)

Maximum number of collection IDs to return.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 4096.

Required: No

NextToken (p. 157)

Pagination token from the previous response.

Type: String

Length Constraints: Maximum length of 255.

Required: No

Response Syntax

```
{  
  "CollectionIds": [ "string" ],  
  "FaceModelVersions": [ "string" ],  
  "NextToken": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[CollectionIds \(p. 157\)](#)

An array of collection IDs.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\-]+

[FaceModelVersions \(p. 157\)](#)

Version numbers of the face detection models associated with the collections in the array `CollectionIds`. For example, the value of `FaceModelVersions[2]` is the version number for the face detection model used by the collection in `CollectionId[2]`.

Type: Array of strings

[NextToken \(p. 157\)](#)

If the result is truncated, the response provides a `NextToken` that you can use in the subsequent request to fetch the next set of collection IDs.

Type: String

Length Constraints: Maximum length of 255.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListDatasetEntries

Lists the entries (images) within a dataset. An entry is a JSON Line that contains the information for a single image, including the image location, assigned labels, and object location bounding boxes. For more information, see [Creating a manifest file](#).

JSON Lines in the response include information about non-terminal errors found in the dataset. Non terminal errors are reported in `errors` lists within each JSON Line. The same information is reported in the training and testing validation result manifests that Amazon Rekognition Custom Labels creates during model training.

You can filter the response in variety of ways, such as choosing which labels to return and returning JSON Lines created after a specific date.

This operation requires permissions to perform the `rekognition:ListDatasetEntries` action.

Request Syntax

```
{  
  "ContainsLabels": [ "string" ],  
  "DatasetArn": "string",  
  "HasErrors": boolean,  
  "Labeled": boolean,  
  "MaxResults": number,  
  "NextToken": "string",  
  "SourceRefContains": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ContainsLabels \(p. 160\)](#)

Specifies a label filter for the response. The response includes an entry only if one or more of the labels in `ContainsLabels` exist in the entry.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: .{1,}

Required: No

[DatasetArn \(p. 160\)](#)

The Amazon Resource Name (ARN) for the dataset that you want to use.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:project\/[a-zA-Z0-9_.\-\-]\{1,255\}\dataset\/(train|test)\/[0-9]+\\$)

Required: Yes

[HasErrors \(p. 160\)](#)

Specifies an error filter for the response. Specify True to only include entries that have errors.

Type: Boolean

Required: No

[Labeled \(p. 160\)](#)

Specify true to get only the JSON Lines where the image is labeled. Specify false to get only the JSON Lines where the image isn't labeled. If you don't specify Labeled, ListDatasetEntries returns JSON Lines for labeled and unlabeled images.

Type: Boolean

Required: No

[MaxResults \(p. 160\)](#)

The maximum number of results to return per paginated call. The largest value you can specify is 100. If you specify a value greater than 100, a ValidationException error occurs. The default value is 100.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

[NextToken \(p. 160\)](#)

If the previous response was incomplete (because there is more results to retrieve), Amazon Rekognition Custom Labels returns a pagination token in the response. You can use this pagination token to retrieve the next set of results.

Type: String

Length Constraints: Maximum length of 1024.

Required: No

[SourceRefContains \(p. 160\)](#)

If specified, ListDatasetEntries only returns JSON Lines where the value of SourceRefContains is part of the source-ref field. The source-ref field contains the Amazon S3 location of the image. You can use SourceRefContains for tasks such as getting the JSON Line for a single image, or getting JSON Lines for all images within a specific folder.

For more information, see [Creating a manifest file](#).

For more information, see [Creating a manifest file in the Amazon Rekognition Custom Labels Developer Guide](#).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: .*\\$.*

Required: No

Response Syntax

```
{  
  "DatasetEntries": [ "string" ],  
  "NextToken": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[DatasetEntries \(p. 162\)](#)

A list of entries (images) in the dataset.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100000.

Pattern: ^\{.*\}\$

[NextToken \(p. 162\)](#)

If the previous response was incomplete (because there is more results to retrieve), Amazon Rekognition Custom Labels returns a pagination token in the response. You can use this pagination token to retrieve the next set of results.

Type: String

Length Constraints: Maximum length of 1024.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ResourceNotReadyException

The requested resource isn't ready. For example, this exception occurs when you call DetectCustomLabels with a model version that isn't deployed.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListDatasetLabels

Lists the labels in a dataset. Amazon Rekognition Custom Labels uses labels to describe images. For more information, see [Labeling images](#).

This operation requires permissions to perform the `rekognition:ListDatasetLabels` action.

Request Syntax

```
{  
  "DatasetArn": "string",  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[DatasetArn \(p. 164\)](#)

The Amazon Resource Name (ARN) of the dataset that you want to use.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:project\/[a-zA-Z0-9_.\-\d]{1,255}\dataset\/(train|test)\/[0-9]+\$)

Required: Yes

[MaxResults \(p. 164\)](#)

The maximum number of results to return per paginated call. The largest value you can specify is 100. If you specify a value greater than 100, a `ValidationException` error occurs. The default value is 100.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

[NextToken \(p. 164\)](#)

If the previous response was incomplete (because there is more results to retrieve), Amazon Rekognition Custom Labels returns a pagination token in the response. You can use this pagination token to retrieve the next set of results.

Type: String

Length Constraints: Maximum length of 1024.

Required: No

Response Syntax

```
{  
  "DatasetLabelDescriptions": [  
    {  
      "LabelName": "string",  
      "LabelStats": {  
        "BoundingBoxCount": number,  
        "EntryCount": number  
      }  
    }  
  ],  
  "NextToken": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[DatasetLabelDescriptions \(p. 165\)](#)

A list of the labels in the dataset.

Type: Array of [DatasetLabelDescription \(p. 278\)](#) objects

[NextToken \(p. 165\)](#)

If the previous response was incomplete (because there is more results to retrieve), Amazon Rekognition Custom Labels returns a pagination token in the response. You can use this pagination token to retrieve the next set of results.

Type: String

Length Constraints: Maximum length of 1024.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ResourceNotReadyException

The requested resource isn't ready. For example, this exception occurs when you call DetectCustomLabels with a model version that isn't deployed.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListFaces

Returns metadata for faces in the specified collection. This metadata includes information such as the bounding box coordinates, the confidence (that the bounding box contains a face), and face ID. For an example, see [Listing faces in a collection](#).

This operation requires permissions to perform the `rekognition:ListFaces` action.

Request Syntax

```
{  
  "CollectionId": "string",  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[CollectionId \(p. 167\)](#)

ID of the collection from which to list the faces.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

[MaxResults \(p. 167\)](#)

Maximum number of faces to return.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 4096.

Required: No

[NextToken \(p. 167\)](#)

If the previous response was incomplete (because there is more data to retrieve), Amazon Rekognition returns a pagination token in the response. You can use this pagination token to retrieve the next set of faces.

Type: String

Length Constraints: Maximum length of 255.

Required: No

Response Syntax

```
{  
  "FaceModelVersion": "string",  
  "Faces": [  
    {  
      "BoundingBox": {  
        "Height": number,  
        "Left": number,  
        "Top": number,  
        "Width": number  
      },  
      "Confidence": number,  
      "ExternalImageId": "string",  
      "FaceId": "string",  
      "ImageId": "string",  
      "IndexFacesModelVersion": "string"  
    }  
  ],  
  "NextToken": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

FaceModelVersion (p. 168)

Version number of the face detection model associated with the input collection (CollectionId).

Type: String

Faces (p. 168)

An array of Face objects.

Type: Array of [Face](#) (p. 300) objects

NextToken (p. 168)

If the response is truncated, Amazon Rekognition returns this token that you can use in the subsequent request to retrieve the next set of faces.

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 391).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListProjectPolicies

Gets a list of the project policies attached to a project.

To attach a project policy to a project, call [PutProjectPolicy \(p. 179\)](#). To remove a project policy from a project, call [DeleteProjectPolicy \(p. 47\)](#).

Request Syntax

```
{  
  "MaxResults": number,  
  "NextToken": string,  
  "ProjectArn": string  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

MaxResults (p. 170)

The maximum number of results to return per paginated call. The largest value you can specify is 5. If you specify a value greater than 5, a `ValidationException` error occurs. The default value is 5.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 5.

Required: No

NextToken (p. 170)

If the previous response was incomplete (because there is more results to retrieve), Amazon Rekognition Custom Labels returns a pagination token in the response. You can use this pagination token to retrieve the next set of results.

Type: String

Length Constraints: Maximum length of 1024.

Required: No

ProjectArn (p. 170)

The ARN of the project for which you want to list the project policies.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\:\d{12}:project\/[a-zA-Z0-9_.\-\-]\{1,255\}\/[0-9]+\\$)

Required: Yes

Response Syntax

```
{  
  "NextToken": "string",  
  "ProjectPolicies": [  
    {  
      "CreationTimestamp": number,  
      "LastUpdatedTimestamp": number,  
      "PolicyDocument": "string",  
      "PolicyName": "string",  
      "PolicyRevisionId": "string",  
      "ProjectArn": "string"  
    }  
  ]  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

NextToken (p. 171)

If the response is truncated, Amazon Rekognition returns this token that you can use in the subsequent request to retrieve the next set of project policies.

Type: String

Length Constraints: Maximum length of 1024.

ProjectPolicies (p. 171)

A list of project policies attached to the project.

Type: Array of [ProjectPolicy](#) (p. 342) objects

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 391).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidOperationException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListStreamProcessors

Gets a list of stream processors that you have created with [CreateStreamProcessor \(p. 31\)](#).

Request Syntax

```
{  
  "MaxResults": number,  
  "NextToken": string  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[MaxResults \(p. 173\)](#)

Maximum number of stream processors you want Amazon Rekognition Video to return in the response. The default is 1000.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

[NextToken \(p. 173\)](#)

If the previous response was incomplete (because there are more stream processors to retrieve), Amazon Rekognition Video returns a pagination token in the response. You can use this pagination token to retrieve the next set of stream processors.

Type: String

Length Constraints: Maximum length of 255.

Required: No

Response Syntax

```
{  
  "NextToken": string,  
  "StreamProcessors": [  
    {  
      "Name": string,  
      "Status": string  
    }  
  ]  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[NextToken \(p. 173\)](#)

If the response is truncated, Amazon Rekognition Video returns this token that you can use in the subsequent request to retrieve the next set of stream processors.

Type: String

Length Constraints: Maximum length of 255.

[StreamProcessors \(p. 173\)](#)

List of stream processors that you have created.

Type: Array of [StreamProcessor \(p. 367\)](#) objects

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidPaginationTokenException

Pagination token in the request is not valid.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListTagsForResource

Returns a list of tags in an Amazon Rekognition collection, stream processor, or Custom Labels model.

This operation requires permissions to perform the `rekognition:ListTagsForResource` action.

Request Syntax

```
{  
  "ResourceArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ResourceArn \(p. 176\)](#)

Amazon Resource Name (ARN) of the model, collection, or stream processor that contains the tags that you want a list of.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Required: Yes

Response Syntax

```
{  
  "Tags": {  
    "string" : "string"  
  }  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[Tags \(p. 176\)](#)

A list of key-value tags assigned to the resource.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 200 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `^(?!aws:)[\p{L}\p{Z}\p{N}_.:/=+\-@]*$`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: ^([\\p{L}\\p{Z}\\p{N}_.:/=-@]*)\$

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

PutProjectPolicy

Attaches a project policy to a Amazon Rekognition Custom Labels project in a trusting AWS account. A project policy specifies that a trusted AWS account can copy a model version from a trusting AWS account to a project in the trusted AWS account. To copy a model version you use the [CopyProjectVersion \(p. 11\)](#) operation.

For more information about the format of a project policy document, see [Attaching a project policy \(SDK\)](#).

The response from PutProjectPolicy is a revision ID for the project policy. You can attach multiple project policies to a project. You can also update an existing project policy by specifying the policy revision ID of the existing policy.

To remove a project policy from a project, call [DeleteProjectPolicy \(p. 47\)](#). To get a list of project policies attached to a project, call [ListProjectPolicies \(p. 170\)](#).

You copy a model version by calling [CopyProjectVersion \(p. 11\)](#).

Request Syntax

```
{  
  "PolicyDocument": "string",  
  "PolicyName": "string",  
  "PolicyRevisionId": "string",  
  "ProjectArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

PolicyDocument (p. 179)

A resource policy to add to the model. The policy is a JSON structure that contains one or more statements that define the policy. The policy must follow the IAM syntax. For more information about the contents of a JSON policy document, see [IAM JSON policy reference](#).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2000.

Pattern: [\u0009\u000A\u000D\u0020-\u00FF]+

Required: Yes

PolicyName (p. 179)

A name for the policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: [a-zA-Z0-9_.\u002D]+

Required: Yes

[PolicyRevisionId \(p. 179\)](#)

The revision ID for the Project Policy. Each time you modify a policy, Amazon Rekognition Custom Labels generates and assigns a new PolicyRevisionId and then deletes the previous version of the policy.

Type: String

Length Constraints: Maximum length of 64.

Pattern: [0-9A-Fa-f]+

Required: No

[ProjectArn \(p. 179\)](#)

The Amazon Resource Name (ARN) of the project that the project policy is attached to.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-zA-Z\d-]+:rekognition:[a-zA-Z\d-]+:\d{12}:project\[/[a-zA-Z0-9_.\-_]{1,255}\]/[0-9]+\$)

Required: Yes

Response Syntax

```
{  
  "PolicyRevisionId": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[PolicyRevisionId \(p. 180\)](#)

The ID of the project policy.

Type: String

Length Constraints: Maximum length of 64.

Pattern: [0-9A-Fa-f]+

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidPolicyRevisionIdException

The supplied revision id for the project policy is invalid.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a **LimitExceededException** exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

MalformedPolicyDocumentException

The format of the project policy document that you supplied to **PutProjectPolicy** is incorrect.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceAlreadyExistsException

A resource with the specified ID already exists.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ServiceQuotaExceededException

The size of the resource exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

RecognizeCelebrities

Returns an array of celebrities recognized in the input image. For more information, see [Recognizing celebrities](#).

RecognizeCelebrities returns the 64 largest faces in the image. It lists the recognized celebrities in the `CelebrityFaces` array and any unrecognized faces in the `UnrecognizedFaces` array. RecognizeCelebrities doesn't return celebrities whose faces aren't among the largest 64 faces in the image.

For each celebrity recognized, RecognizeCelebrities returns a `Celebrity` object. The `Celebrity` object contains the celebrity name, ID, URL links to additional information, match confidence, and a `ComparedFace` object that you can use to locate the celebrity's face on the image.

Amazon Rekognition doesn't retain information about which images a celebrity has been recognized in. Your application must store this information and use the `Celebrity` ID property as a unique identifier for the celebrity. If you don't store the celebrity name or additional information URLs returned by RecognizeCelebrities, you will need the ID to identify the celebrity in a call to the [GetCelebrityInfo \(p. 105\)](#) operation.

You pass the input image either as base64-encoded image bytes or as a reference to an image in an Amazon S3 bucket. If you use the AWS CLI to call Amazon Rekognition operations, passing image bytes is not supported. The image must be either a PNG or JPEG formatted file.

For an example, see [Recognizing celebrities in an image](#).

This operation requires permissions to perform the `rekognition:RecognizeCelebrities` operation.

Request Syntax

```
{  
  "Image": {  
    "Bytes": blob,  
    "S3Object": {  
      "Bucket": "string",  
      "Name": "string",  
      "Version": "string"  
    }  
  }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[Image \(p. 183\)](#)

The input image as base64-encoded bytes or an S3 object. If you use the AWS CLI to call Amazon Rekognition operations, passing base64-encoded image bytes is not supported.

If you are using an AWS SDK to call Amazon Rekognition, you might not need to base64-encode image bytes passed using the `Bytes` field. For more information, see [Image specifications](#).

Type: [Image \(p. 317\)](#) object

Required: Yes

Response Syntax

```
{
  "CelebrityFaces": [
    {
      "Face": {
        "BoundingBox": {
          "Height": number,
          "Left": number,
          "Top": number,
          "Width": number
        },
        "Confidence": number,
        "Emotions": [
          {
            "Confidence": number,
            "Type": "string"
          }
        ],
        "Landmarks": [
          {
            "Type": "string",
            "X": number,
            "Y": number
          }
        ],
        "Pose": {
          "Pitch": number,
          "Roll": number,
          "Yaw": number
        },
        "Quality": {
          "Brightness": number,
          "Sharpness": number
        },
        "Smile": {
          "Confidence": number,
          "Value": boolean
        }
      },
      "Id": "string",
      "KnownGender": {
        "Type": "string"
      },
      "MatchConfidence": number,
      "Name": "string",
      "Urls": [ "string " ]
    }
  ],
  "OrientationCorrection": "string",
  "UnrecognizedFaces": [
    {
      "BoundingBox": {
        "Height": number,
        "Left": number,
        "Top": number,
        "Width": number
      },
      "Confidence": number,
      "Name": "string"
    }
  ]
}
```

```

    "Emotions": [
      {
        "Confidence": number,
        "Type": "string"
      }
    ],
    "Landmarks": [
      {
        "Type": "string",
        "X": number,
        "Y": number
      }
    ],
    "Pose": {
      "Pitch": number,
      "Roll": number,
      "Yaw": number
    },
    "Quality": {
      "Brightness": number,
      "Sharpness": number
    },
    "Smile": {
      "Confidence": number,
      "Value": boolean
    }
  }
]
}

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[CelebrityFaces \(p. 184\)](#)

Details about each celebrity found in the image. Amazon Rekognition can detect a maximum of 64 celebrities in an image. Each celebrity object includes the following attributes: Face, Confidence, Emotions, Landmarks, Pose, Quality, Smile, Id, KnownGender, MatchConfidence, Name, Urls.

Type: Array of [Celebrity \(p. 261\)](#) objects

[OrientationCorrection \(p. 184\)](#)

Note

Support for estimating image orientation using the the OrientationCorrection field has ceased as of August 2021. Any returned values for this field included in an API response will always be NULL.

The orientation of the input image (counterclockwise direction). If your application displays the image, you can use this value to correct the orientation. The bounding box coordinates returned in [CelebrityFaces](#) and [UnrecognizedFaces](#) represent face locations before the image orientation is corrected.

Note

If the input image is in jpeg format, it might contain exchangeable image (Exif) metadata that includes the image's orientation. If so, and the Exif metadata for the input image populates the orientation field, the value of OrientationCorrection is null. The [CelebrityFaces](#) and [UnrecognizedFaces](#) bounding box coordinates represent face

locations after Exif metadata is used to correct the image orientation. Images in .png format don't contain Exif metadata.

Type: String

Valid Values: ROTATE_0 | ROTATE_90 | ROTATE_180 | ROTATE_270

[UnrecognizedFaces \(p. 184\)](#)

Details about each unrecognized face in the image.

Type: Array of [ComparedFace \(p. 266\)](#) objects

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

ImageTooLargeException

The input image size exceeds the allowed limit. If you are calling [DetectProtectiveEquipment \(p. 95\)](#), the image size or resolution exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidImageFormatException

The provided image format is not supported.

HTTP Status Code: 400

InvalidImageFormatException

The provided image format is not supported.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

SearchFaces

For a given input face ID, searches for matching faces in the collection the face belongs to. You get a face ID when you add a face to the collection using the [IndexFaces \(p. 148\)](#) operation. The operation compares the features of the input face with faces in the specified collection.

Note

You can also search faces without indexing faces by using the [SearchFacesByImage](#) operation.

The operation response returns an array of faces that match, ordered by similarity score with the highest similarity first. More specifically, it is an array of metadata for each face match that is found. Along with the metadata, the response also includes a confidence value for each face match, indicating the confidence that the specific face matches the input face.

For an example, see [Searching for a face using its face ID](#).

This operation requires permissions to perform the `rekognition:SearchFaces` action.

Request Syntax

```
{  
  "CollectionId": "string",  
  "FaceId": "string",  
  "FaceMatchThreshold": number,  
  "MaxFaces": number  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[CollectionId \(p. 188\)](#)

ID of the collection the face belongs to.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

[Facelid \(p. 188\)](#)

ID of a face to find matches for in the collection.

Type: String

Pattern: [0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}

Required: Yes

[FaceMatchThreshold \(p. 188\)](#)

Optional value specifying the minimum confidence in the face match to return. For example, don't return any matches where confidence in matches is less than 70%. The default value is 80%.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

[MaxFaces \(p. 188\)](#)

Maximum number of faces to return. The operation returns the maximum number of faces with the highest confidence in the match.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 4096.

Required: No

Response Syntax

```
{  
  "FaceMatches": [  
    {  
      "Face": {  
        "BoundingBox": {  
          "Height": number,  
          "Left": number,  
          "Top": number,  
          "Width": number  
        },  
        "Confidence": number,  
        "ExternalImageId": "string",  
        "FaceId": "string",  
        "ImageId": "string",  
        "IndexFacesModelVersion": "string"  
      },  
      "Similarity": number  
    }  
  ],  
  "FaceModelVersion": "string",  
  "SearchedFaceId": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[FaceMatches \(p. 189\)](#)

An array of faces that matched the input face, along with the confidence in the match.

Type: Array of [FaceMatch \(p. 306\)](#) objects

[FaceModelVersion \(p. 189\)](#)

Version number of the face detection model associated with the input collection (CollectionId).

Type: String

[SearchedFaceId \(p. 189\)](#)

ID of the face that was searched for matches in a collection.

Type: String

Pattern: [0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

SearchFacesByImage

For a given input image, first detects the largest face in the image, and then searches the specified collection for matching faces. The operation compares the features of the input face with faces in the specified collection.

Note

To search for all faces in an input image, you might first call the [IndexFaces \(p. 148\)](#) operation, and then use the face IDs returned in subsequent calls to the [SearchFaces \(p. 188\)](#) operation.

You can also call the [DetectFaces](#) operation and use the bounding boxes in the response to make face crops, which then you can pass in to the [SearchFacesByImage](#) operation.

You pass the input image either as base64-encoded image bytes or as a reference to an image in an Amazon S3 bucket. If you use the AWS CLI to call Amazon Rekognition operations, passing image bytes is not supported. The image must be either a PNG or JPEG formatted file.

The response returns an array of faces that match, ordered by similarity score with the highest similarity first. More specifically, it is an array of metadata for each face match found. Along with the metadata, the response also includes a [similarity](#) indicating how similar the face is to the input face. In the response, the operation also returns the bounding box (and a confidence level that the bounding box contains a face) of the face that Amazon Rekognition used for the input image.

If no faces are detected in the input image, [SearchFacesByImage](#) returns an [InvalidParameterException](#) error.

For an example, see [Searching for a Face Using an Image](#).

The `QualityFilter` input parameter allows you to filter out detected faces that don't meet a required quality bar. The quality bar is based on a variety of common use cases. Use `QualityFilter` to set the quality bar for filtering by specifying `LOW`, `MEDIUM`, or `HIGH`. If you do not want to filter detected faces, specify `NONE`. The default value is `NONE`.

Note

To use quality filtering, you need a collection associated with version 3 of the face model or higher. To get the version of the face model associated with a collection, call [DescribeCollection \(p. 55\)](#).

This operation requires permissions to perform the `rekognition:SearchFacesByImage` action.

Request Syntax

```
{  
  "CollectionId": "string",  
  "FaceMatchThreshold": number,  
  "Image": {  
    "Bytes": blob,  
    "S3Object": {  
      "Bucket": "string",  
      "Name": "string",  
      "Version": "string"  
    }  
  },  
  "MaxFaces": number,  
  "QualityFilter": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[CollectionId \(p. 192\)](#)

ID of the collection to search.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\+]+

Required: Yes

[FaceMatchThreshold \(p. 192\)](#)

(Optional) Specifies the minimum confidence in the face match to return. For example, don't return any matches where confidence in matches is less than 70%. The default value is 80%.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

[Image \(p. 192\)](#)

The input image as base64-encoded bytes or an S3 object. If you use the AWS CLI to call Amazon Rekognition operations, passing base64-encoded image bytes is not supported.

If you are using an AWS SDK to call Amazon Rekognition, you might not need to base64-encode image bytes passed using the Bytes field. For more information, see [Image specifications](#).

Type: [Image \(p. 317\)](#) object

Required: Yes

[MaxFaces \(p. 192\)](#)

Maximum number of faces to return. The operation returns the maximum number of faces with the highest confidence in the match.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 4096.

Required: No

[QualityFilter \(p. 192\)](#)

A filter that specifies a quality bar for how much filtering is done to identify faces. Filtered faces aren't searched for in the collection. If you specify AUTO, Amazon Rekognition chooses the quality bar. If you specify LOW, MEDIUM, or HIGH, filtering removes all faces that don't meet the chosen quality bar. The quality bar is based on a variety of common use cases. Low-quality detections can occur for a number of reasons. Some examples are an object that's misidentified as a face, a face that's too blurry, or a face with a pose that's too extreme to use. If you specify NONE, no filtering is performed. The default value is NONE.

To use quality filtering, the collection you are using must be associated with version 3 of the face model or higher.

Type: String

Valid Values: NONE | AUTO | LOW | MEDIUM | HIGH

Required: No

Response Syntax

```
{  
  "FaceMatches": [  
    {  
      "Face": {  
        "BoundingBox": {  
          "Height": number,  
          "Left": number,  
          "Top": number,  
          "Width": number  
        },  
        "Confidence": number,  
        "ExternalImageId": "string",  
        "FaceId": "string",  
        "ImageId": "string",  
        "IndexFacesModelVersion": "string"  
      },  
      "Similarity": number  
    }  
  ],  
  "FaceModelVersion": "string",  
  "SearchedFaceBoundingBox": {  
    "Height": number,  
    "Left": number,  
    "Top": number,  
    "Width": number  
  },  
  "SearchedFaceConfidence": number  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[FaceMatches \(p. 194\)](#)

An array of faces that match the input face, along with the confidence in the match.

Type: Array of [FaceMatch \(p. 306\)](#) objects

[FaceModelVersion \(p. 194\)](#)

Version number of the face detection model associated with the input collection (CollectionId).

Type: String

[SearchedFaceBoundingBox \(p. 194\)](#)

The bounding box around the face in the input image that Amazon Rekognition used for the search.

Type: [BoundingBox \(p. 259\)](#) object

SearchedFaceConfidence (p. 194)

The level of confidence that the `searchedFaceBoundingBox` contains a face.

Type: `Float`

Valid Range: Minimum value of 0. Maximum value of 100.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

ImageTooLargeException

The input image size exceeds the allowed limit. If you are calling [DetectProtectiveEquipment \(p. 95\)](#), the image size or resolution exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidImageFormatException

The provided image format is not supported.

HTTP Status Code: 400

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartCelebrityRecognition

Starts asynchronous recognition of celebrities in a stored video.

Amazon Rekognition Video can detect celebrities in a video must be stored in an Amazon S3 bucket. Use [Video \(p. 386\)](#) to specify the bucket name and the filename of the video. StartCelebrityRecognition returns a job identifier (JobId) which you use to get the results of the analysis. When celebrity recognition analysis is finished, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic that you specify in [NotificationChannel](#). To get the results of the celebrity recognition analysis, first check that the status value published to the Amazon SNS topic is SUCCEEDED. If so, call [GetCelebrityRecognition \(p. 108\)](#) and pass the job identifier (JobId) from the initial call to StartCelebrityRecognition.

For more information, see [Recognizing celebrities](#).

Request Syntax

```
{  
  "ClientRequestToken": "string",  
  "JobTag": "string",  
  "NotificationChannel": {  
    "RoleArn": "string",  
    "SNSTopicArn": "string"  
  },  
  "Video": {  
    "S3Object": {  
      "Bucket": "string",  
      "Name": "string",  
      "Version": "string"  
    }  
  }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ClientRequestToken \(p. 197\)](#)

Idempotent token used to identify the start request. If you use the same token with multiple StartCelebrityRecognition requests, the same JobId is returned. Use ClientRequestToken to prevent the same job from being accidentally started more than once.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: No

[JobTag \(p. 197\)](#)

An identifier you specify that's returned in the completion notification that's published to your Amazon Simple Notification Service topic. For example, you can use JobTag to group related jobs and identify them in the completion notification.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9_.\-\:]+

Required: No

[NotificationChannel \(p. 197\)](#)

The Amazon SNS topic ARN that you want Amazon Rekognition Video to publish the completion status of the celebrity recognition analysis to. The Amazon SNS topic must have a topic name that begins with *AmazonRekognition* if you are using the *AmazonRekognitionServiceRole* permissions policy.

Type: [NotificationChannel \(p. 333\)](#) object

Required: No

[Video \(p. 197\)](#)

The video in which you want to recognize celebrities. The video must be stored in an Amazon S3 bucket.

Type: [Video \(p. 386\)](#) object

Required: Yes

Response Syntax

```
{  
  "JobId": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[JobId \(p. 198\)](#)

The identifier for the celebrity recognition analysis job. Use *JobId* to identify the job in a subsequent call to *GetCelebrityRecognition*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

IdempotentParameterMismatchException

A ClientRequestToken input parameter was reused with an operation, but at least one of the other input parameters is different from the previous call to the operation.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a LimitExceededException exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

VideoTooLargeException

The file size or duration of the supplied media is too large. The maximum file size is 10GB. The maximum duration is 6 hours.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)

- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartContentModeration

Starts asynchronous detection of inappropriate, unwanted, or offensive content in a stored video. For a list of moderation labels in Amazon Rekognition, see [Using the image and video moderation APIs](#).

Amazon Rekognition Video can moderate content in a video stored in an Amazon S3 bucket. Use [Video \(p. 386\)](#) to specify the bucket name and the filename of the video. StartContentModeration returns a job identifier (JobId) which you use to get the results of the analysis. When content analysis is finished, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic that you specify in NotificationChannel.

To get the results of the content analysis, first check that the status value published to the Amazon SNS topic is SUCCEEDED. If so, call [GetContentModeration \(p. 114\)](#) and pass the job identifier (JobId) from the initial call to StartContentModeration.

For more information, see [Moderating content](#).

Request Syntax

```
{  
    "ClientRequestToken": "string",  
    "JobTag": "string",  
    "MinConfidence": number,  
    "NotificationChannel": {  
        "RoleArn": "string",  
        "SNSTopicArn": "string"  
    },  
    "Video": {  
        "S3Object": {  
            "Bucket": "string",  
            "Name": "string",  
            "Version": "string"  
        }  
    }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ClientRequestToken \(p. 201\)](#)

Idempotent token used to identify the start request. If you use the same token with multiple StartContentModeration requests, the same JobId is returned. Use ClientRequestToken to prevent the same job from being accidentally started more than once.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: No

[JobTag \(p. 201\)](#)

An identifier you specify that's returned in the completion notification that's published to your Amazon Simple Notification Service topic. For example, you can use JobTag to group related jobs and identify them in the completion notification.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9_.\-\:]+

Required: No

[MinConfidence \(p. 201\)](#)

Specifies the minimum confidence that Amazon Rekognition must have in order to return a moderated content label. Confidence represents how certain Amazon Rekognition is that the moderated content is correctly identified. 0 is the lowest confidence. 100 is the highest confidence. Amazon Rekognition doesn't return any moderated content labels with a confidence level lower than this specified value. If you don't specify MinConfidence, GetContentModeration returns labels with confidence values greater than or equal to 50 percent.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

[NotificationChannel \(p. 201\)](#)

The Amazon SNS topic ARN that you want Amazon Rekognition Video to publish the completion status of the content analysis to. The Amazon SNS topic must have a topic name that begins with *AmazonRekognition* if you are using the *AmazonRekognitionServiceRole* permissions policy to access the topic.

Type: [NotificationChannel \(p. 333\)](#) object

Required: No

[Video \(p. 201\)](#)

The video in which you want to detect inappropriate, unwanted, or offensive content. The video must be stored in an Amazon S3 bucket.

Type: [Video \(p. 386\)](#) object

Required: Yes

Response Syntax

```
{  
  "JobId": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

JobId (p. 202)

The identifier for the content analysis job. Use `JobId` to identify the job in a subsequent call to `GetContentModeration`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z0-9-_]+$`

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

IdempotentParameterMismatchException

A `ClientRequestToken` input parameter was reused with an operation, but at least one of the other input parameters is different from the previous call to the operation.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (`StartLabelDetection`, for example) will raise a `LimitExceededException` exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

VideoTooLargeException

The file size or duration of the supplied media is too large. The maximum file size is 10GB. The maximum duration is 6 hours.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartFaceDetection

Starts asynchronous detection of faces in a stored video.

Amazon Rekognition Video can detect faces in a video stored in an Amazon S3 bucket. Use [Video \(p. 386\)](#) to specify the bucket name and the filename of the video. StartFaceDetection returns a job identifier (JobId) that you use to get the results of the operation. When face detection is finished, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic that you specify in NotificationChannel. To get the results of the face detection operation, first check that the status value published to the Amazon SNS topic is SUCCEEDED. If so, call [GetFaceDetection \(p. 118\)](#) and pass the job identifier (JobId) from the initial call to StartFaceDetection.

For more information, see [Detecting faces in a stored video](#).

Request Syntax

```
{  
    "ClientRequestToken": "string",  
    "FaceAttributes": "string",  
    "JobTag": "string",  
    "NotificationChannel": {  
        "RoleArn": "string",  
        "SNSTopicArn": "string"  
    },  
    "Video": {  
        "S3Object": {  
            "Bucket": "string",  
            "Name": "string",  
            "Version": "string"  
        }  
    }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ClientRequestToken \(p. 205\)](#)

Idempotent token used to identify the start request. If you use the same token with multiple StartFaceDetection requests, the same JobId is returned. Use ClientRequestToken to prevent the same job from being accidentally started more than once.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: No

[FaceAttributes \(p. 205\)](#)

The face attributes you want returned.

DEFAULT - The following subset of facial attributes are returned: BoundingBox, Confidence, Pose, Quality and Landmarks.

ALL - All facial attributes are returned.

Type: String

Valid Values: DEFAULT | ALL

Required: No

[JobTag \(p. 205\)](#)

An identifier you specify that's returned in the completion notification that's published to your Amazon Simple Notification Service topic. For example, you can use JobTag to group related jobs and identify them in the completion notification.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9_.\-\:]+

Required: No

[NotificationChannel \(p. 205\)](#)

The ARN of the Amazon SNS topic to which you want Amazon Rekognition Video to publish the completion status of the face detection operation. The Amazon SNS topic must have a topic name that begins with *AmazonRekognition* if you are using the *AmazonRekognitionServiceRole* permissions policy.

Type: [NotificationChannel \(p. 333\)](#) object

Required: No

[Video \(p. 205\)](#)

The video in which you want to detect faces. The video must be stored in an Amazon S3 bucket.

Type: [Video \(p. 386\)](#) object

Required: Yes

Response Syntax

```
{  
  "JobId": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[JobId \(p. 206\)](#)

The identifier for the face detection job. Use *JobId* to identify the job in a subsequent call to *GetFaceDetection*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

IdempotentParameterMismatchException

A ClientRequestToken input parameter was reused with an operation, but at least one of the other input parameters is different from the previous call to the operation.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a LimitExceededException exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

VideoTooLargeException

The file size or duration of the supplied media is too large. The maximum file size is 10GB. The maximum duration is 6 hours.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartFaceSearch

Starts the asynchronous search for faces in a collection that match the faces of persons detected in a stored video.

The video must be stored in an Amazon S3 bucket. Use [Video \(p. 386\)](#) to specify the bucket name and the filename of the video. StartFaceSearch returns a job identifier (JobId) which you use to get the search results once the search has completed. When searching is finished, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic that you specify in `NotificationChannel1`. To get the search results, first check that the status value published to the Amazon SNS topic is `SUCCEEDED`. If so, call [GetFaceSearch \(p. 123\)](#) and pass the job identifier (JobId) from the initial call to StartFaceSearch. For more information, see [Searching stored videos for faces](#).

Request Syntax

```
{  
  "ClientRequestToken": "string",  
  "CollectionId": "string",  
  "FaceMatchThreshold": number,  
  "JobTag": "string",  
  "NotificationChannel": {  
    "RoleArn": "string",  
    "SNSTopicArn": "string"  
  },  
  "Video": {  
    "S3Object": {  
      "Bucket": "string",  
      "Name": "string",  
      "Version": "string"  
    }  
  }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ClientRequestToken \(p. 209\)](#)

Idempotent token used to identify the start request. If you use the same token with multiple StartFaceSearch requests, the same JobId is returned. Use `ClientRequestToken` to prevent the same job from being accidentally started more than once.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z0-9-_]+$`

Required: No

[CollectionId \(p. 209\)](#)

ID of the collection that contains the faces you want to search for.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

[FaceMatchThreshold \(p. 209\)](#)

The minimum confidence in the person match to return. For example, don't return any matches where confidence in matches is less than 70%. The default value is 80%.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

[JobTag \(p. 209\)](#)

An identifier you specify that's returned in the completion notification that's published to your Amazon Simple Notification Service topic. For example, you can use JobTag to group related jobs and identify them in the completion notification.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: No

[NotificationChannel \(p. 209\)](#)

The ARN of the Amazon SNS topic to which you want Amazon Rekognition Video to publish the completion status of the search. The Amazon SNS topic must have a topic name that begins with *AmazonRekognition* if you are using the *AmazonRekognitionServiceRole* permissions policy to access the topic.

Type: [NotificationChannel \(p. 333\)](#) object

Required: No

[Video \(p. 209\)](#)

The video you want to search. The video must be stored in an Amazon S3 bucket.

Type: [Video \(p. 386\)](#) object

Required: Yes

Response Syntax

```
{  
  "JobId": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

JobId (p. 210)

The identifier for the search job. Use `JobId` to identify the job in a subsequent call to `GetFaceSearch`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z0-9-_]+$`

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

IdempotentParameterMismatchException

A `ClientRequestToken` input parameter was reused with an operation, but at least one of the other input parameters is different from the previous call to the operation.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (`StartLabelDetection`, for example) will raise a `LimitExceededException` exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

VideoTooLargeException

The file size or duration of the supplied media is too large. The maximum file size is 10GB. The maximum duration is 6 hours.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartLabelDetection

Starts asynchronous detection of labels in a stored video.

Amazon Rekognition Video can detect labels in a video. Labels are instances of real-world entities. This includes objects like flower, tree, and table; events like wedding, graduation, and birthday party; concepts like landscape, evening, and nature; and activities like a person getting out of a car or a person skiing.

The video must be stored in an Amazon S3 bucket. Use [Video \(p. 386\)](#) to specify the bucket name and the filename of the video. StartLabelDetection returns a job identifier (JobId) which you use to get the results of the operation. When label detection is finished, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic that you specify in [NotificationChannel](#).

To get the results of the label detection operation, first check that the status value published to the Amazon SNS topic is SUCCEEDED. If so, call [GetLabelDetection \(p. 129\)](#) and pass the job identifier (JobId) from the initial call to StartLabelDetection.

Request Syntax

```
{  
  "ClientRequestToken": "string",  
  "JobTag": "string",  
  "MinConfidence": number,  
  "NotificationChannel": {  
    "RoleArn": "string",  
    "SNSTopicArn": "string"  
  },  
  "Video": {  
    "S3Object": {  
      "Bucket": "string",  
      "Name": "string",  
      "Version": "string"  
    }  
  }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ClientRequestToken \(p. 213\)](#)

Idempotent token used to identify the start request. If you use the same token with multiple StartLabelDetection requests, the same JobId is returned. Use ClientRequestToken to prevent the same job from being accidentally started more than once.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: No

[JobTag \(p. 213\)](#)

An identifier you specify that's returned in the completion notification that's published to your Amazon Simple Notification Service topic. For example, you can use JobTag to group related jobs and identify them in the completion notification.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9_.\-\:]+

Required: No

[MinConfidence \(p. 213\)](#)

Specifies the minimum confidence that Amazon Rekognition Video must have in order to return a detected label. Confidence represents how certain Amazon Rekognition is that a label is correctly identified. 0 is the lowest confidence. 100 is the highest confidence. Amazon Rekognition Video doesn't return any labels with a confidence level lower than this specified value.

If you don't specify MinConfidence, the operation returns labels with confidence values greater than or equal to 50 percent.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

[NotificationChannel \(p. 213\)](#)

The Amazon SNS topic ARN you want Amazon Rekognition Video to publish the completion status of the label detection operation to. The Amazon SNS topic must have a topic name that begins with *AmazonRekognition* if you are using the *AmazonRekognitionServiceRole* permissions policy.

Type: [NotificationChannel \(p. 333\)](#) object

Required: No

[Video \(p. 213\)](#)

The video in which you want to detect labels. The video must be stored in an Amazon S3 bucket.

Type: [Video \(p. 386\)](#) object

Required: Yes

Response Syntax

```
{  
  "JobId": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

JobId (p. 214)

The identifier for the label detection job. Use `JobId` to identify the job in a subsequent call to `GetLabelDetection`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z0-9-_]+$`

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

IdempotentParameterMismatchException

A `ClientRequestToken` input parameter was reused with an operation, but at least one of the other input parameters is different from the previous call to the operation.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (`StartLabelDetection`, for example) will raise a `LimitExceededException` exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

VideoTooLargeException

The file size or duration of the supplied media is too large. The maximum file size is 10GB. The maximum duration is 6 hours.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartPersonTracking

Starts the asynchronous tracking of a person's path in a stored video.

Amazon Rekognition Video can track the path of people in a video stored in an Amazon S3 bucket. Use [Video \(p. 386\)](#) to specify the bucket name and the filename of the video. `StartPersonTracking` returns a job identifier (`JobId`) which you use to get the results of the operation. When label detection is finished, Amazon Rekognition publishes a completion status to the Amazon Simple Notification Service topic that you specify in `NotificationChannel`.

To get the results of the person detection operation, first check that the status value published to the Amazon SNS topic is `SUCCEEDED`. If so, call [GetPersonTracking \(p. 134\)](#) and pass the job identifier (`JobId`) from the initial call to `StartPersonTracking`.

Request Syntax

```
{  
  "ClientRequestToken": "string",  
  "JobTag": "string",  
  "NotificationChannel": {  
    "RoleArn": "string",  
    "SNSTopicArn": "string"  
  },  
  "Video": {  
    "S3Object": {  
      "Bucket": "string",  
      "Name": "string",  
      "Version": "string"  
    }  
  }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ClientRequestToken \(p. 217\)](#)

Idempotent token used to identify the start request. If you use the same token with multiple `StartPersonTracking` requests, the same `JobId` is returned. Use `ClientRequestToken` to prevent the same job from being accidentally started more than once.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z0-9-_]+$`

Required: No

[JobTag \(p. 217\)](#)

An identifier you specify that's returned in the completion notification that's published to your Amazon Simple Notification Service topic. For example, you can use `JobTag` to group related jobs and identify them in the completion notification.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9_.\-\:]\+

Required: No

[NotificationChannel \(p. 217\)](#)

The Amazon SNS topic ARN you want Amazon Rekognition Video to publish the completion status of the people detection operation to. The Amazon SNS topic must have a topic name that begins with *AmazonRekognition* if you are using the *AmazonRekognitionServiceRole* permissions policy.

Type: [NotificationChannel \(p. 333\)](#) object

Required: No

[Video \(p. 217\)](#)

The video in which you want to detect people. The video must be stored in an Amazon S3 bucket.

Type: [Video \(p. 386\)](#) object

Required: Yes

Response Syntax

```
{  
  "JobId": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[JobId \(p. 218\)](#)

The identifier for the person detection job. Use *JobId* to identify the job in a subsequent call to *GetPersonTracking*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]\+\$

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

IdempotentParameterMismatchException

A ClientRequestToken input parameter was reused with an operation, but at least one of the other input parameters is different from the previous call to the operation.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a LimitExceededException exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

VideoTooLargeException

The file size or duration of the supplied media is too large. The maximum file size is 10GB. The maximum duration is 6 hours.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartProjectVersion

Starts the running of the version of a model. Starting a model takes a while to complete. To check the current state of the model, use [DescribeProjectVersions \(p. 64\)](#).

Once the model is running, you can detect custom labels in new images by calling [DetectCustomLabels \(p. 74\)](#).

Note

You are charged for the amount of time that the model is running. To stop a running model, call [StopProjectVersion \(p. 235\)](#).

For more information, see [Running a trained Amazon Rekognition Custom Labels model](#).

This operation requires permissions to perform the `rekognition:StartProjectVersion` action.

Request Syntax

```
{  
  "MaxInferenceUnits": number,  
  "MinInferenceUnits": number,  
  "ProjectVersionArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[MaxInferenceUnits \(p. 221\)](#)

The maximum number of inference units to use for auto-scaling the model. If you don't specify a value, Amazon Rekognition Custom Labels doesn't auto-scale the model.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

[MinInferenceUnits \(p. 221\)](#)

The minimum number of inference units to use. A single inference unit represents 1 hour of processing.

For information about the number of transactions per second (TPS) that an inference unit can support, see [Running a trained Amazon Rekognition Custom Labels model](#).

Use a higher number to increase the TPS throughput of your model. You are charged for the number of inference units that you use.

Type: Integer

Valid Range: Minimum value of 1.

Required: Yes

[ProjectVersionArn \(p. 221\)](#)

The Amazon Resource Name(ARN) of the model version that you want to start.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\version\/[a-zA-Z0-9_.\-\-]{1,255}\[0-9]+\$)

Required: Yes

Response Syntax

```
{  
  "Status": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[Status \(p. 222\)](#)

The current running status of the model.

Type: String

Valid Values: TRAINING_IN_PROGRESS | TRAINING_COMPLETED | TRAINING_FAILED | STARTING | RUNNING | FAILED | STOPPING | STOPPED | DELETING | COPYING_IN_PROGRESS | COPYING_COMPLETED | COPYING_FAILED

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a `LimitExceededException` exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartSegmentDetection

Starts asynchronous detection of segment detection in a stored video.

Amazon Rekognition Video can detect segments in a video stored in an Amazon S3 bucket. Use [Video \(p. 386\)](#) to specify the bucket name and the filename of the video. `StartSegmentDetection` returns a job identifier (`JobId`) which you use to get the results of the operation. When segment detection is finished, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic that you specify in `NotificationChannel`.

You can use the `Filters` ([StartSegmentDetectionFilters \(p. 361\)](#)) input parameter to specify the minimum detection confidence returned in the response. Within `Filters`, use `ShotFilter` ([StartShotDetectionFilter \(p. 362\)](#)) to filter detected shots. Use `TechnicalCueFilter` ([StartTechnicalCueDetectionFilter \(p. 363\)](#)) to filter technical cues.

To get the results of the segment detection operation, first check that the status value published to the Amazon SNS topic is `SUCCEEDED`. If so, call [GetSegmentDetection \(p. 139\)](#) and pass the job identifier (`JobId`) from the initial call to `StartSegmentDetection`.

For more information, see [Detecting video segments in stored video](#).

Request Syntax

```
{  
    "ClientRequestToken": "string",  
    "Filters": {  
        "ShotFilter": {  
            "MinSegmentConfidence": number  
        },  
        "TechnicalCueFilter": {  
            "BlackFrame": {  
                "MaxPixelThreshold": number,  
                "MinCoveragePercentage": number  
            },  
            "MinSegmentConfidence": number  
        }  
    },  
    "JobTag": "string",  
    "NotificationChannel": {  
        "RoleArn": "string",  
        "SNSTopicArn": "string"  
    },  
    "SegmentTypes": [ "string" ],  
    "Video": {  
        "S3Object": {  
            "Bucket": "string",  
            "Name": "string",  
            "Version": "string"  
        }  
    }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ClientRequestToken \(p. 224\)](#)

Idempotent token used to identify the start request. If you use the same token with multiple StartSegmentDetection requests, the same JobId is returned. Use ClientRequestToken to prevent the same job from being accidentally started more than once.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: No

[Filters \(p. 224\)](#)

Filters for technical cue or shot detection.

Type: [StartSegmentDetectionFilters \(p. 361\)](#) object

Required: No

[JobTag \(p. 224\)](#)

An identifier you specify that's returned in the completion notification that's published to your Amazon Simple Notification Service topic. For example, you can use JobTag to group related jobs and identify them in the completion notification.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9_.\-\:]+

Required: No

[NotificationChannel \(p. 224\)](#)

The ARN of the Amazon SNS topic to which you want Amazon Rekognition Video to publish the completion status of the segment detection operation. Note that the Amazon SNS topic must have a topic name that begins with *AmazonRekognition* if you are using the *AmazonRekognitionServiceRole* permissions policy to access the topic.

Type: [NotificationChannel \(p. 333\)](#) object

Required: No

[SegmentTypes \(p. 224\)](#)

An array of segment types to detect in the video. Valid values are TECHNICAL_CUE and SHOT.

Type: Array of strings

Array Members: Minimum number of 1 item.

Valid Values: TECHNICAL_CUE | SHOT

Required: Yes

[Video \(p. 224\)](#)

Video file stored in an Amazon S3 bucket. Amazon Rekognition video start operations such as [StartLabelDetection \(p. 213\)](#) use Video to specify a video for analysis. The supported file formats are .mp4, .mov and .avi.

Type: [Video \(p. 386\)](#) object

Required: Yes

Response Syntax

```
{  
  "JobId": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

JobId (p. 226)

Unique identifier for the segment detection job. The JobId is returned from StartSegmentDetection.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

IdempotentParameterMismatchException

A ClientRequestToken input parameter was reused with an operation, but at least one of the other input parameters is different from the previous call to the operation.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a LimitExceededException exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

VideoTooLargeException

The file size or duration of the supplied media is too large. The maximum file size is 10GB. The maximum duration is 6 hours.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartStreamProcessor

Starts processing a stream processor. You create a stream processor by calling [CreateStreamProcessor \(p. 31\)](#). To tell StartStreamProcessor which stream processor to start, use the value of the Name field specified in the call to CreateStreamProcessor.

If you are using a label detection stream processor to detect labels, you need to provide a Start selector and a Stop selector to determine the length of the stream processing time.

Request Syntax

```
{  
  "Name": "string",  
  "StartSelector": {  
    "KVSSStreamStartSelector": {  
      "FragmentNumber": "string",  
      "ProducerTimestamp": number  
    }  
  },  
  "StopSelector": {  
    "MaxDurationInSeconds": number  
  }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[Name \(p. 228\)](#)

The name of the stream processor to start processing.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

[StartSelector \(p. 228\)](#)

Specifies the starting point in the Kinesis stream to start processing. You can use the producer timestamp or the fragment number. If you use the producer timestamp, you must put the time in milliseconds. For more information about fragment numbers, see [Fragment](#).

This is a required parameter for label detection stream processors and should not be used to start a face search stream processor.

Type: [StreamProcessingStartSelector \(p. 365\)](#) object

Required: No

[StopSelector \(p. 228\)](#)

Specifies when to stop processing the stream. You can specify a maximum amount of time to process the video.

This is a required parameter for label detection stream processors and should not be used to start a face search stream processor.

Type: [StreamProcessingStopSelector \(p. 366\)](#) object

Required: No

Response Syntax

```
{  
  "SessionId": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

SessionId (p. 229)

A unique identifier for the stream processing session.

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartTextDetection

Starts asynchronous detection of text in a stored video.

Amazon Rekognition Video can detect text in a video stored in an Amazon S3 bucket. Use [Video \(p. 386\)](#) to specify the bucket name and the filename of the video. `StartTextDetection` returns a job identifier (`JobId`) which you use to get the results of the operation. When text detection is finished, Amazon Rekognition Video publishes a completion status to the Amazon Simple Notification Service topic that you specify in `NotificationChannel`.

To get the results of the text detection operation, first check that the status value published to the Amazon SNS topic is `SUCCEEDED`. If so, call [GetTextDetection \(p. 144\)](#) and pass the job identifier (`JobId`) from the initial call to `StartTextDetection`.

Request Syntax

```
{  
    "ClientRequestToken": "string",  
    "Filters": {  
        "RegionsOfInterest": [  
            {  
                "BoundingBox": {  
                    "Height": number,  
                    "Left": number,  
                    "Top": number,  
                    "Width": number  
                },  
                "Polygon": [  
                    {  
                        "X": number,  
                        "Y": number  
                    }  
                ]  
            }  
        ],  
        "WordFilter": {  
            "MinBoundingBoxHeight": number,  
            "MinBoundingBoxWidth": number,  
            "MinConfidence": number  
        }  
    },  
    "JobTag": "string",  
    "NotificationChannel": {  
        "RoleArn": "string",  
        "SNSTopicArn": "string"  
    },  
    "Video": {  
        "S3Object": {  
            "Bucket": "string",  
            "Name": "string",  
            "Version": "string"  
        }  
    }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ClientRequestToken \(p. 231\)](#)

Idempotent token used to identify the start request. If you use the same token with multiple StartTextDetection requests, the same JobId is returned. Use ClientRequestToken to prevent the same job from being accidentally started more than once.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z0-9-_]+\$

Required: No

[Filters \(p. 231\)](#)

Optional parameters that let you set criteria the text must meet to be included in your response.

Type: [StartTextDetectionFilters \(p. 364\)](#) object

Required: No

[JobTag \(p. 231\)](#)

An identifier returned in the completion status published by your Amazon Simple Notification Service topic. For example, you can use JobTag to group related jobs and identify them in the completion notification.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9_.\-\:]+

Required: No

[NotificationChannel \(p. 231\)](#)

The Amazon Simple Notification Service topic to which Amazon Rekognition publishes the completion status of a video analysis operation. For more information, see [Calling Amazon Rekognition Video operations](#). Note that the Amazon SNS topic must have a topic name that begins with *AmazonRekognition* if you are using the *AmazonRekognitionServiceRole* permissions policy to access the topic. For more information, see [Giving access to multiple Amazon SNS topics](#).

Type: [NotificationChannel \(p. 333\)](#) object

Required: No

[Video \(p. 231\)](#)

Video file stored in an Amazon S3 bucket. Amazon Rekognition video start operations such as [StartLabelDetection \(p. 213\)](#) use Video to specify a video for analysis. The supported file formats are .mp4, .mov and .avi.

Type: [Video \(p. 386\)](#) object

Required: Yes

Response Syntax

```
{
```

```
    "JobId": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

JobId (p. 232)

Identifier for the text detection job. Use `JobId` to identify the job in a subsequent call to `GetTextDetection`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z0-9-_]+$`

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

IdempotentParameterMismatchException

A `ClientRequestToken` input parameter was reused with an operation, but at least one of the other input parameters is different from the previous call to the operation.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

InvalidS3ObjectException

Amazon Rekognition is unable to access the S3 object specified in the request.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (`StartLabelDetection`, for example) will raise a `LimitExceededException` exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

VideoTooLargeException

The file size or duration of the supplied media is too large. The maximum file size is 10GB. The maximum duration is 6 hours.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StopProjectVersion

Stops a running model. The operation might take a while to complete. To check the current status, call [DescribeProjectVersions \(p. 64\)](#).

Request Syntax

```
{  
  "ProjectVersionArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ProjectVersionArn \(p. 235\)](#)

The Amazon Resource Name (ARN) of the model version that you want to delete.

This operation requires permissions to perform the `rekognition:StopProjectVersion` action.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\version\/[a-zA-Z0-9_.\-\-]{1,255}\[0-9]+\$)

Required: Yes

Response Syntax

```
{  
  "Status": "string"  
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

[Status \(p. 235\)](#)

The current status of the stop operation.

Type: String

Valid Values: TRAINING_IN_PROGRESS | TRAINING_COMPLETED | TRAINING_FAILED | STARTING | RUNNING | FAILED | STOPPING | STOPPED | DELETING | COPYING_IN_PROGRESS | COPYING_COMPLETED | COPYING_FAILED

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)

- [AWS SDK for Ruby V3](#)

StopStreamProcessor

Stops a running stream processor that was created by [CreateStreamProcessor \(p. 31\)](#).

Request Syntax

```
{  
    "Name": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

Name (p. 238)

The name of a stream processor created by [CreateStreamProcessor \(p. 31\)](#).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

TagResource

Adds one or more key-value tags to an Amazon Rekognition collection, stream processor, or Custom Labels model. For more information, see [Tagging AWS Resources](#).

This operation requires permissions to perform the `rekognition:TagResource` action.

Request Syntax

```
{  
  "ResourceArn": "string",  
  "Tags": {  
    "string" : "string"  
  }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

ResourceArn (p. 240)

Amazon Resource Name (ARN) of the model, collection, or stream processor that you want to assign the tags to.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Required: Yes

Tags (p. 240)

The key-value tags to assign to the resource.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 200 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `^(?!aws:)[\p{L}\p{Z}]\p{N}_.:/=+\-@]*$`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `^([\p{L}\p{Z}]\p{N}_.:/=+\-@)*$`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ServiceQuotaExceededException

The size of the resource exceeds the allowed limit. For more information, see [Guidelines and quotas in Amazon Rekognition](#).

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

UntagResource

Removes one or more tags from an Amazon Rekognition collection, stream processor, or Custom Labels model.

This operation requires permissions to perform the `rekognition:UntagResource` action.

Request Syntax

```
{  
  "ResourceArn": "string",  
  "TagKeys": [ "string" ]  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[ResourceArn \(p. 243\)](#)

Amazon Resource Name (ARN) of the model, collection, or stream processor that you want to remove the tags from.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Required: Yes

[TagKeys \(p. 243\)](#)

A list of the tags that you want to remove.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 200 items.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `^(!aws:)[\p{L}\p{Z}\p{N}_.:/+=\-\@]*$`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

UpdateDatasetEntries

Adds or updates one or more entries (images) in a dataset. An entry is a JSON Line which contains the information for a single image, including the image location, assigned labels, and object location bounding boxes. For more information, see [Image-Level labels in manifest files](#) and [Object localization in manifest files](#).

If the `source-ref` field in the JSON line references an existing image, the existing image in the dataset is updated. If `source-ref` field doesn't reference an existing image, the image is added as a new image to the dataset.

You specify the changes that you want to make in the `Changes` input parameter. There isn't a limit to the number JSON Lines that you can change, but the size of `Changes` must be less than 5MB.

`UpdateDatasetEntries` returns immediately, but the dataset update might take a while to complete. Use [DescribeDataset \(p. 58\)](#) to check the current status. The dataset updated successfully if the value of `Status` is `UPDATE_COMPLETE`.

To check if any non-terminal errors occurred, call [ListDatasetEntries \(p. 160\)](#) and check for the presence of `errors` lists in the JSON Lines.

Dataset update fails if a terminal error occurs (`Status` = `UPDATE_FAILED`). Currently, you can't access the terminal error information from the Amazon Rekognition Custom Labels SDK.

This operation requires permissions to perform the `rekognition:UpdateDatasetEntries` action.

Request Syntax

```
{  
  "Changes": {  
    "GroundTruth": blob  
  },  
  "DatasetArn": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[Changes \(p. 245\)](#)

The changes that you want to make to the dataset.

Type: [DatasetChanges \(p. 275\)](#) object

Required: Yes

[DatasetArn \(p. 245\)](#)

The Amazon Resource Name (ARN) of the dataset that you want to update.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\:\d{12}:project\/[a-zA-Z0-9_.\-\-]\{1,255\}\dataset\/(train|test)\/[0-9]+\\$)

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidParameterException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

LimitExceededException

An Amazon Rekognition service limit was exceeded. For example, if you start too many Amazon Rekognition Video jobs concurrently, calls to start operations (StartLabelDetection, for example) will raise a LimitExceededException exception (HTTP status code: 400) until the number of concurrently running jobs is below the Amazon Rekognition service limit.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceInUseException

The specified resource is already being used.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

UpdateStreamProcessor

Allows you to update a stream processor. You can change some settings and regions of interest and delete certain parameters.

Request Syntax

```
{  
  "DataSharingPreferenceForUpdate": {  
    "OptIn": boolean  
  },  
  "Name": "string",  
  "ParametersToDelete": [ "string" ],  
  "RegionsOfInterestForUpdate": [  
    {  
      "BoundingBox": {  
        "Height": number,  
        "Left": number,  
        "Top": number,  
        "Width": number  
      },  
      "Polygon": [  
        {  
          "X": number,  
          "Y": number  
        }  
      ]  
    }  
  ],  
  "SettingsForUpdate": {  
    "ConnectedHomeForUpdate": {  
      "Labels": [ "string" ],  
      "MinConfidence": number  
    }  
  }  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 389\)](#).

The request accepts the following data in JSON format.

[DataSharingPreferenceForUpdate \(p. 248\)](#)

Shows whether you are sharing data with Rekognition to improve model performance. You can choose this option at the account level or on a per-stream basis. Note that if you opt out at the account level this setting is ignored on individual streams.

Type: [StreamProcessorDataSharingPreference \(p. 368\)](#) object

Required: No

[Name \(p. 248\)](#)

Name of the stream processor that you want to update.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: Yes

[ParametersToDelete \(p. 248\)](#)

A list of parameters you want to delete from the stream processor.

Type: Array of strings

Valid Values: ConnectedHomeMinConfidence | RegionsOfInterest

Required: No

[RegionsOfInterestForUpdate \(p. 248\)](#)

Specifies locations in the frames where Amazon Rekognition checks for objects or people. This is an optional parameter for label detection stream processors.

Type: Array of [RegionOfInterest \(p. 352\)](#) objects

Array Members: Minimum number of 0 items. Maximum number of 10 items.

Required: No

[SettingsForUpdate \(p. 248\)](#)

The stream processor settings that you want to update. Label detection settings can be updated to detect different labels with a different minimum confidence.

Type: [StreamProcessorSettingsForUpdate \(p. 373\)](#) object

Required: No

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 391\)](#).

AccessDeniedException

You are not authorized to perform the action.

HTTP Status Code: 400

InternalServerError

Amazon Rekognition experienced a service issue. Try your call again.

HTTP Status Code: 500

InvalidArgumentException

Input parameter violated a constraint. Validate your parameter before calling the API operation again.

HTTP Status Code: 400

ProvisionedThroughputExceededException

The number of requests exceeded your throughput limit. If you want to increase this limit, contact Amazon Rekognition.

HTTP Status Code: 400

ResourceNotFoundException

The resource specified in the request cannot be found.

HTTP Status Code: 400

ThrottlingException

Amazon Rekognition is temporarily unable to process the request. Try your call again.

HTTP Status Code: 500

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

Data Types

The Amazon Rekognition API contains several data types that various actions use. This section describes each data type in detail.

Note

The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- [AgeRange \(p. 254\)](#)
- [Asset \(p. 255\)](#)
- [AudioMetadata \(p. 256\)](#)
- [Beard \(p. 257\)](#)
- [BlackFrame \(p. 258\)](#)
- [BoundingBox \(p. 259\)](#)
- [Celebrity \(p. 261\)](#)
- [CelebrityDetail \(p. 263\)](#)
- [CelebrityRecognition \(p. 265\)](#)
- [ComparedFace \(p. 266\)](#)
- [ComparedSourceImageFace \(p. 268\)](#)
- [CompareFacesMatch \(p. 269\)](#)
- [ConnectedHomeSettings \(p. 270\)](#)
- [ConnectedHomeSettingsForUpdate \(p. 271\)](#)
- [ContentModerationDetection \(p. 272\)](#)
- [CoversBodyPart \(p. 273\)](#)
- [CustomLabel \(p. 274\)](#)
- [DatasetChanges \(p. 275\)](#)
- [DatasetDescription \(p. 276\)](#)
- [DatasetLabelDescription \(p. 278\)](#)
- [DatasetLabelStats \(p. 279\)](#)
- [DatasetMetadata \(p. 280\)](#)
- [DatasetSource \(p. 282\)](#)
- [DatasetStats \(p. 283\)](#)
- [DetectionFilter \(p. 284\)](#)
- [DetectLabelsImageBackground \(p. 285\)](#)
- [DetectLabelsImageForeground \(p. 286\)](#)
- [DetectLabelsImageProperties \(p. 287\)](#)
- [DetectLabelsImagePropertiesSettings \(p. 288\)](#)
- [DetectLabelsImageQuality \(p. 289\)](#)
- [DetectLabelsSettings \(p. 290\)](#)
- [DetectTextFilters \(p. 291\)](#)
- [DistributeDataset \(p. 292\)](#)
- [DominantColor \(p. 293\)](#)
- [Emotion \(p. 295\)](#)

- [EquipmentDetection \(p. 296\)](#)
- [EvaluationResult \(p. 297\)](#)
- [Eyeglasses \(p. 298\)](#)
- [EyeOpen \(p. 299\)](#)
- [Face \(p. 300\)](#)
- [FaceDetail \(p. 302\)](#)
- [FaceDetection \(p. 305\)](#)
- [FaceMatch \(p. 306\)](#)
- [FaceRecord \(p. 307\)](#)
- [FaceSearchSettings \(p. 308\)](#)
- [Gender \(p. 309\)](#)
- [GeneralLabelsSettings \(p. 310\)](#)
- [Geometry \(p. 312\)](#)
- [GroundTruthManifest \(p. 313\)](#)
- [HumanLoopActivationOutput \(p. 314\)](#)
- [HumanLoopConfig \(p. 315\)](#)
- [HumanLoopDataAttributes \(p. 316\)](#)
- [Image \(p. 317\)](#)
- [ImageQuality \(p. 318\)](#)
- [Instance \(p. 319\)](#)
- [KinesisDataStream \(p. 320\)](#)
- [KinesisVideoStream \(p. 321\)](#)
- [KinesisVideoStreamStartSelector \(p. 322\)](#)
- [KnownGender \(p. 323\)](#)
- [Label \(p. 324\)](#)
- [LabelAlias \(p. 326\)](#)
- [LabelCategory \(p. 327\)](#)
- [LabelDetection \(p. 328\)](#)
- [Landmark \(p. 329\)](#)
- [ModerationLabel \(p. 330\)](#)
- [MouthOpen \(p. 331\)](#)
- [Mustache \(p. 332\)](#)
- [NotificationChannel \(p. 333\)](#)
- [OutputConfig \(p. 334\)](#)
- [Parent \(p. 335\)](#)
- [PersonDetail \(p. 336\)](#)
- [PersonDetection \(p. 337\)](#)
- [PersonMatch \(p. 338\)](#)
- [Point \(p. 339\)](#)
- [Pose \(p. 340\)](#)
- [ProjectDescription \(p. 341\)](#)
- [ProjectPolicy \(p. 342\)](#)
- [ProjectVersionDescription \(p. 344\)](#)
- [ProtectiveEquipmentBodyPart \(p. 347\)](#)
- [ProtectiveEquipmentPerson \(p. 348\)](#)
- [ProtectiveEquipmentSummarizationAttributes \(p. 349\)](#)

- [ProtectiveEquipmentSummary \(p. 350\)](#)
- [RegionOfInterest \(p. 352\)](#)
- [S3Destination \(p. 353\)](#)
- [S3Object \(p. 354\)](#)
- [SegmentDetection \(p. 355\)](#)
- [SegmentTypeInfo \(p. 358\)](#)
- [ShotSegment \(p. 359\)](#)
- [Smile \(p. 360\)](#)
- [StartSegmentDetectionFilters \(p. 361\)](#)
- [StartShotDetectionFilter \(p. 362\)](#)
- [StartTechnicalCueDetectionFilter \(p. 363\)](#)
- [StartTextDetectionFilters \(p. 364\)](#)
- [StreamProcessingStartSelector \(p. 365\)](#)
- [StreamProcessingStopSelector \(p. 366\)](#)
- [StreamProcessor \(p. 367\)](#)
- [StreamProcessorDataSharingPreference \(p. 368\)](#)
- [StreamProcessorInput \(p. 369\)](#)
- [StreamProcessorNotificationChannel \(p. 370\)](#)
- [StreamProcessorOutput \(p. 371\)](#)
- [StreamProcessorSettings \(p. 372\)](#)
- [StreamProcessorSettingsForUpdate \(p. 373\)](#)
- [Summary \(p. 374\)](#)
- [Sunglasses \(p. 375\)](#)
- [TechnicalCueSegment \(p. 376\)](#)
- [TestingData \(p. 377\)](#)
- [TestingDataResult \(p. 378\)](#)
- [TextDetection \(p. 379\)](#)
- [TextDetectionResult \(p. 381\)](#)
- [TrainingData \(p. 382\)](#)
- [TrainingDataResult \(p. 383\)](#)
- [UnindexedFace \(p. 384\)](#)
- [ValidationData \(p. 385\)](#)
- [Video \(p. 386\)](#)
- [VideoMetadata \(p. 387\)](#)

AgeRange

Structure containing the estimated age range, in years, for a face.

Amazon Rekognition estimates an age range for faces detected in the input image. Estimated age ranges can overlap. A face of a 5-year-old might have an estimated range of 4-6, while the face of a 6-year-old might have an estimated range of 4-8.

Contents

High

The highest estimated age.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

Low

The lowest estimated age.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Asset

Assets are the images that you use to train and evaluate a model version. Assets can also contain validation information that you use to debug a failed model training.

Contents

GroundTruthManifest

The S3 bucket that contains an Amazon Sagemaker Ground Truth format manifest file.

Type: [GroundTruthManifest \(p. 313\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AudioMetadata

Metadata information about an audio stream. An array of AudioMetadata objects for the audio streams found in a stored video is returned by [GetSegmentDetection \(p. 139\)](#).

Contents

Codec

The audio codec used to encode or decode the audio stream.

Type: String

Required: No

DurationMillis

The duration of the audio stream in milliseconds.

Type: Long

Valid Range: Minimum value of 0.

Required: No

NumberOfChannels

The number of audio channels in the segment.

Type: Long

Valid Range: Minimum value of 0.

Required: No

SampleRate

The sample rate for the audio stream.

Type: Long

Valid Range: Minimum value of 0.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Beard

Indicates whether or not the face has a beard, and the confidence level in the determination.

Contents

Confidence

Level of confidence in the determination.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Value

Boolean value that indicates whether the face has beard or not.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

BlackFrame

A filter that allows you to control the black frame detection by specifying the black levels and pixel coverage of black pixels in a frame. As videos can come from multiple sources, formats, and time periods, they may contain different standards and varying noise levels for black frames that need to be accounted for. For more information, see [StartSegmentDetection \(p. 224\)](#).

Contents

MaxPixelThreshold

A threshold used to determine the maximum luminance value for a pixel to be considered black. In a full color range video, luminance values range from 0-255. A pixel value of 0 is pure black, and the most strict filter. The maximum black pixel value is computed as follows: $\text{max_black_pixel_value} = \text{minimum_luminance} + \text{MaxPixelThreshold} * \text{luminance_range}$.

For example, for a full range video with $\text{BlackPixelThreshold} = 0.1$, $\text{max_black_pixel_value}$ is $0 + 0.1 * (255-0) = 25.5$.

The default value of MaxPixelThreshold is 0.2, which maps to a $\text{max_black_pixel_value}$ of 51 for a full range video. You can lower this threshold to be more strict on black levels.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 1.

Required: No

MinCoveragePercentage

The minimum percentage of pixels in a frame that need to have a luminance below the $\text{max_black_pixel_value}$ for a frame to be considered a black frame. Luminance is calculated using the BT.709 matrix.

The default value is 99, which means at least 99% of all pixels in the frame are black pixels as per the MaxPixelThreshold set. You can reduce this value to allow more noise on the black frame.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

BoundingBox

Identifies the bounding box around the label, face, text, object of interest, or personal protective equipment. The left (x-coordinate) and top (y-coordinate) are coordinates representing the top and left sides of the bounding box. Note that the upper-left corner of the image is the origin (0,0).

The top and left values returned are ratios of the overall image size. For example, if the input image is 700x200 pixels, and the top-left coordinate of the bounding box is 350x50 pixels, the API returns a left value of 0.5 (350/700) and a top value of 0.25 (50/200).

The width and height values represent the dimensions of the bounding box as a ratio of the overall image dimension. For example, if the input image is 700x200 pixels, and the bounding box width is 70 pixels, the width returned is 0.1.

Note

The bounding box coordinates can have negative values. For example, if Amazon Rekognition is able to detect a face that is at the image edge and is only partially visible, the service can return coordinates that are outside the image bounds and, depending on the image edge, you might get negative values or values greater than 1 for the left or top values.

Contents

Height

Height of the bounding box as a ratio of the overall image height.

Type: Float

Required: No

Left

Left coordinate of the bounding box as a ratio of overall image width.

Type: Float

Required: No

Top

Top coordinate of the bounding box as a ratio of overall image height.

Type: Float

Required: No

Width

Width of the bounding box as a ratio of the overall image width.

Type: Float

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Celebrity

Provides information about a celebrity recognized by the [RecognizeCelebrities \(p. 183\)](#) operation.

Contents

Face

Provides information about the celebrity's face, such as its location on the image.

Type: [ComparedFace \(p. 266\)](#) object

Required: No

Id

A unique identifier for the celebrity.

Type: String

Pattern: [0-9A-Za-z]*

Required: No

KnownGender

The known gender identity for the celebrity that matches the provided ID. The known gender identity can be Male, Female, Nonbinary, or Unlisted.

Type: [KnownGender \(p. 323\)](#) object

Required: No

MatchConfidence

The confidence, in percentage, that Amazon Rekognition has that the recognized face is the celebrity.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Name

The name of the celebrity.

Type: String

Required: No

Urls

An array of URLs pointing to additional information about the celebrity. If there is no additional information about the celebrity, this list is empty.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 255 items.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

CelebrityDetail

Information about a recognized celebrity.

Contents

BoundingBox

Bounding box around the body of a celebrity.

Type: [BoundingBox \(p. 259\)](#) object

Required: No

Confidence

The confidence, in percentage, that Amazon Rekognition has that the recognized face is the celebrity.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Face

Face details for the recognized celebrity.

Type: [FaceDetail \(p. 302\)](#) object

Required: No

Id

The unique identifier for the celebrity.

Type: String

Pattern: [0-9A-Za-z]*

Required: No

KnownGender

Retrieves the known gender for the celebrity.

Type: [KnownGender \(p. 323\)](#) object

Required: No

Name

The name of the celebrity.

Type: String

Required: No

Urls

An array of URLs pointing to additional celebrity information.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 255 items.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

CelebrityRecognition

Information about a detected celebrity and the time the celebrity was detected in a stored video. For more information, see [GetCelebrityRecognition \(p. 108\)](#).

Contents

Celebrity

Information about a recognized celebrity.

Type: [CelebrityDetail \(p. 263\)](#) object

Required: No

Timestamp

The time, in milliseconds from the start of the video, that the celebrity was recognized. Note that `Timestamp` is not guaranteed to be accurate to the individual frame where the celebrity first appears.

Type: Long

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ComparedFace

Provides face metadata for target image faces that are analyzed by CompareFaces and RecognizeCelebrities.

Contents

BoundingBox

Bounding box of the face.

Type: [BoundingBox \(p. 259\)](#) object

Required: No

Confidence

Level of confidence that what the bounding box contains is a face.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Emotions

The emotions that appear to be expressed on the face, and the confidence level in the determination. Valid values include "Happy", "Sad", "Angry", "Confused", "Disgusted", "Surprised", "Calm", "Unknown", and "Fear".

Type: Array of [Emotion \(p. 295\)](#) objects

Required: No

Landmarks

An array of facial landmarks.

Type: Array of [Landmark \(p. 329\)](#) objects

Required: No

Pose

Indicates the pose of the face as determined by its pitch, roll, and yaw.

Type: [Pose \(p. 340\)](#) object

Required: No

Quality

Identifies face image brightness and sharpness.

Type: [ImageQuality \(p. 318\)](#) object

Required: No

Smile

Indicates whether or not the face is smiling, and the confidence level in the determination.

Type: [Smile \(p. 360\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ComparedSourceImageFace

Type that describes the face Amazon Rekognition chose to compare with the faces in the target. This contains a bounding box for the selected face and confidence level that the bounding box contains a face. Note that Amazon Rekognition selects the largest face in the source image for this comparison.

Contents

BoundingBox

Bounding box of the face.

Type: [BoundingBox \(p. 259\)](#) object

Required: No

Confidence

Confidence level that the selected bounding box contains a face.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

CompareFacesMatch

Provides information about a face in a target image that matches the source image face analyzed by `CompareFaces`. The `Face` property contains the bounding box of the face in the target image. The `Similarity` property is the confidence that the source image face matches the face in the bounding box.

Contents

Face

Provides face metadata (bounding box and confidence that the bounding box actually contains a face).

Type: [ComparedFace \(p. 266\)](#) object

Required: No

Similarity

Level of confidence that the faces match.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ConnectedHomeSettings

Label detection settings to use on a streaming video. Defining the settings is required in the request parameter for [CreateStreamProcessor \(p. 31\)](#). Including this setting in the CreateStreamProcessor request enables you to use the stream processor for label detection. You can then select what you want the stream processor to detect, such as people or pets. When the stream processor has started, one notification is sent for each object class specified. For example, if packages and pets are selected, one SNS notification is published the first time a package is detected and one SNS notification is published the first time a pet is detected, as well as an end-of-session summary.

Contents

Labels

Specifies what you want to detect in the video, such as people, packages, or pets. The current valid labels you can include in this list are: "PERSON", "PET", "PACKAGE", and "ALL".

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 128 items.

Required: Yes

MinConfidence

The minimum confidence required to label an object in the video.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ConnectedHomeSettingsForUpdate

The label detection settings you want to use in your stream processor. This includes the labels you want the stream processor to detect and the minimum confidence level allowed to label objects.

Contents

Labels

Specifies what you want to detect in the video, such as people, packages, or pets. The current valid labels you can include in this list are: "PERSON", "PET", "PACKAGE", and "ALL".

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 128 items.

Required: No

MinConfidence

The minimum confidence required to label an object in the video.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ContentModerationDetection

Information about an inappropriate, unwanted, or offensive content label detection in a stored video.

Contents

ModerationLabel

The content moderation label detected by in the stored video.

Type: [ModerationLabel \(p. 330\)](#) object

Required: No

Timestamp

Time, in milliseconds from the beginning of the video, that the content moderation label was detected. Note that **Timestamp** is not guaranteed to be accurate to the individual frame where the moderated content first appears.

Type: Long

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

CoversBodyPart

Information about an item of Personal Protective Equipment covering a corresponding body part. For more information, see [DetectProtectiveEquipment \(p. 95\)](#).

Contents

Confidence

The confidence that Amazon Rekognition has in the value of **Value**.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Value

True if the PPE covers the corresponding body part, otherwise false.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

CustomLabel

A custom label detected in an image by a call to [DetectCustomLabels \(p. 74\)](#).

Contents

Confidence

The confidence that the model has in the detection of the custom label. The range is 0-100. A higher value indicates a higher confidence.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Geometry

The location of the detected object on the image that corresponds to the custom label. Includes an axis aligned coarse bounding box surrounding the object and a finer grain polygon for more accurate spatial information.

Type: [Geometry \(p. 312\)](#) object

Required: No

Name

The name of the custom label.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DatasetChanges

Describes updates or additions to a dataset. A Single update or addition is an entry (JSON Line) that provides information about a single image. To update an existing entry, you match the `source-ref` field of the update entry with the `source-ref` field of the entry that you want to update. If the `source-ref` field doesn't match an existing entry, the entry is added to dataset as a new entry.

Contents

GroundTruth

A Base64-encoded binary data object containing one or JSON lines that either update the dataset or are additions to the dataset. You change a dataset by calling [UpdateDatasetEntries \(p. 245\)](#). If you are using an AWS SDK to call `UpdateDatasetEntries`, you don't need to encode Changes as the SDK encodes the data for you.

For example JSON lines, see [Image-Level labels in manifest files](#) and [Object localization in manifest files](#).

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 1. Maximum length of 5242880.

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DatasetDescription

A description for a dataset. For more information, see [DescribeDataset \(p. 58\)](#).

The status fields `Status`, `StatusMessage`, and `StatusMessageCode` reflect the last operation on the dataset.

Contents

CreationTimestamp

The Unix timestamp for the time and date that the dataset was created.

Type: `Timestamp`

Required: No

DatasetStats

The status message code for the dataset.

Type: [DatasetStats \(p. 283\)](#) object

Required: No

LastUpdatedTimestamp

The Unix timestamp for the date and time that the dataset was last updated.

Type: `Timestamp`

Required: No

Status

The status of the dataset.

Type: `String`

Valid Values: `CREATE_IN_PROGRESS` | `CREATE_COMPLETE` | `CREATE_FAILED` |
`UPDATE_IN_PROGRESS` | `UPDATE_COMPLETE` | `UPDATE_FAILED` | `DELETE_IN_PROGRESS`

Required: No

StatusMessage

The status message for the dataset.

Type: `String`

Required: No

StatusMessageCode

The status message code for the dataset operation. If a service error occurs, try the API call again later. If a client error occurs, check the input parameters to the dataset API call that failed.

Type: `String`

Valid Values: `SUCCESS` | `SERVICE_ERROR` | `CLIENT_ERROR`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DatasetLabelDescription

Describes a dataset label. For more information, see [ListDatasetLabels \(p. 164\)](#).

Contents

LabelName

The name of the label.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: .{1,}

Required: No

LabelStats

Statistics about the label.

Type: [DatasetLabelStats \(p. 279\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DatasetLabelStats

Statistics about a label used in a dataset. For more information, see [DatasetLabelDescription \(p. 278\)](#).

Contents

BoundingBoxCount

The total number of images that have the label assigned to a bounding box.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

EntryCount

The total number of images that use the label.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DatasetMetadata

Summary information for an Amazon Rekognition Custom Labels dataset. For more information, see [ProjectDescription \(p. 341\)](#).

Contents

CreationTimestamp

The Unix timestamp for the date and time that the dataset was created.

Type: Timestamp

Required: No

DatasetArn

The Amazon Resource Name (ARN) for the dataset.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:project\/[a-zA-Z0-9_.\-\]{1,255}\dataset\/(train|test)\/[0-9]+\\$)

Required: No

DatasetType

The type of the dataset.

Type: String

Valid Values: TRAIN | TEST

Required: No

Status

The status for the dataset.

Type: String

Valid Values: CREATE_IN_PROGRESS | CREATE_COMPLETE | CREATE_FAILED | UPDATE_IN_PROGRESS | UPDATE_COMPLETE | UPDATE_FAILED | DELETE_IN_PROGRESS

Required: No

StatusMessage

The status message for the dataset.

Type: String

Required: No

StatusMessageCode

The status message code for the dataset operation. If a service error occurs, try the API call again later. If a client error occurs, check the input parameters to the dataset API call that failed.

Type: String

Valid Values: SUCCESS | SERVICE_ERROR | CLIENT_ERROR

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DatasetSource

The source that Amazon Rekognition Custom Labels uses to create a dataset. To use an Amazon Sagemaker format manifest file, specify the S3 bucket location in the `GroundTruthManifest` field. The S3 bucket must be in your AWS account. To create a copy of an existing dataset, specify the Amazon Resource Name (ARN) of an existing dataset in `DatasetArn`.

You need to specify a value for `DatasetArn` or `GroundTruthManifest`, but not both. If you supply both values, or if you don't specify any values, an `InvalidParameterException` exception occurs.

For more information, see [CreateDataset \(p. 19\)](#).

Contents

DatasetArn

The ARN of an Amazon Rekognition Custom Labels dataset that you want to copy.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `(^arn:[a-z\d-]+:rekognition:[a-z\d-]+:\d{12}:project\/[a-zA-Z0-9_.\-\]{1,255}\dataset\/(train|test)\/[0-9]+$)`

Required: No

GroundTruthManifest

The S3 bucket that contains an Amazon Sagemaker Ground Truth format manifest file.

Type: [GroundTruthManifest \(p. 313\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DatasetStats

Provides statistics about a dataset. For more information, see [DescribeDataset \(p. 58\)](#).

Contents

ErrorEntries

The total number of entries that contain at least one error.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

LabeledEntries

The total number of images in the dataset that have labels.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

TotalEntries

The total number of images in the dataset.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

TotalLabels

The total number of labels declared in the dataset.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DetectionFilter

A set of parameters that allow you to filter out certain results from your returned results.

Contents

MinBoundingBoxHeight

Sets the minimum height of the word bounding box. Words with bounding box heights lesser than this value will be excluded from the result. Value is relative to the video frame height.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 1.

Required: No

MinBoundingBoxWidth

Sets the minimum width of the word bounding box. Words with bounding boxes widths lesser than this value will be excluded from the result. Value is relative to the video frame width.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 1.

Required: No

MinConfidence

Sets the confidence of word detection. Words with detection confidence below this will be excluded from the result. Values should be between 0 and 100. The default MinConfidence is 80.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DetectLabelsImageBackground

The background of the image with regard to image quality and dominant colors.

Contents

DominantColors

The dominant colors found in the background of an image, defined with RGB values, CSS color name, simplified color name, and PixelPercentage (the percentage of image pixels that have a particular color).

Type: Array of [DominantColor \(p. 293\)](#) objects

Required: No

Quality

The quality of the image background as defined by brightness and sharpness.

Type: [DetectLabelsImageQuality \(p. 289\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DetectLabelsImageForeground

The foreground of the image with regard to image quality and dominant colors.

Contents

DominantColors

The dominant colors found in the foreground of an image, defined with RGB values, CSS color name, simplified color name, and PixelPercentage (the percentage of image pixels that have a particular color).

Type: Array of [DominantColor \(p. 293\)](#) objects

Required: No

Quality

The quality of the image foreground as defined by brightness and sharpness.

Type: [DetectLabelsImageQuality \(p. 289\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DetectLabelsImageProperties

Information about the quality and dominant colors of an input image. Quality and color information is returned for the entire image, foreground, and background.

Contents

Background

Information about the properties of an image's background, including the background's quality and dominant colors, including the quality and dominant colors of the image.

Type: [DetectLabelsImageBackground \(p. 285\)](#) object

Required: No

DominantColors

Information about the dominant colors found in an image, described with RGB values, CSS color name, simplified color name, and PixelPercentage (the percentage of image pixels that have a particular color).

Type: Array of [DominantColor \(p. 293\)](#) objects

Required: No

Foreground

Information about the properties of an image's foreground, including the foreground's quality and dominant colors, including the quality and dominant colors of the image.

Type: [DetectLabelsImageForeground \(p. 286\)](#) object

Required: No

Quality

Information about the quality of the image foreground as defined by brightness, sharpness, and contrast. The higher the value the greater the brightness, sharpness, and contrast respectively.

Type: [DetectLabelsImageQuality \(p. 289\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DetectLabelsImagePropertiesSettings

Settings for the IMAGE_PROPERTIES feature type.

Contents

MaxDominantColors

The maximum number of dominant colors to return when detecting labels in an image. The default value is 10.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 20.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DetectLabelsImageQuality

The quality of an image provided for label detection, with regard to brightness, sharpness, and contrast.

Contents

Brightness

The brightness of an image provided for label detection.

Type: Float

Required: No

Contrast

The contrast of an image provided for label detection.

Type: Float

Required: No

Sharpness

The sharpness of an image provided for label detection.

Type: Float

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DetectLabelsSettings

Settings for the DetectLabels request. Settings can include filters for both GENERAL_LABELS and IMAGE_PROPERTIES. GENERAL_LABELS filters can be inclusive or exclusive and applied to individual labels or label categories. IMAGE_PROPERTIES filters allow specification of a maximum number of dominant colors.

Contents

GeneralLabels

Contains the specified filters for GENERAL_LABELS.

Type: [GeneralLabelsSettings \(p. 310\)](#) object

Required: No

ImageProperties

Contains the chosen number of maximum dominant colors in an image.

Type: [DetectLabelsImagePropertiesSettings \(p. 288\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DetectTextFilters

A set of optional parameters that you can use to set the criteria that the text must meet to be included in your response. `WordFilter` looks at a word's height, width, and minimum confidence. `RegionOfInterest` lets you set a specific region of the image to look for text in.

Contents

RegionsOfInterest

A Filter focusing on a certain area of the image. Uses a `BoundingBox` object to set the region of the image.

Type: Array of [RegionOfInterest \(p. 352\)](#) objects

Array Members: Minimum number of 0 items. Maximum number of 10 items.

Required: No

WordFilter

A set of parameters that allow you to filter out certain results from your returned results.

Type: [DetectionFilter \(p. 284\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DistributeDataset

A training dataset or a test dataset used in a dataset distribution operation. For more information, see [DistributeDatasetEntries \(p. 103\)](#).

Contents

Arn

The Amazon Resource Name (ARN) of the dataset that you want to use.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+:\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\dataset\/(train|test)\/[0-9]+\\$)

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DominantColor

A description of the dominant colors in an image.

Contents

Blue

The Blue RGB value for a dominant color.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

CSSColor

The CSS color name of a dominant color.

Type: String

Required: No

Green

The Green RGB value for a dominant color.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

HexCode

The Hex code equivalent of the RGB values for a dominant color.

Type: String

Required: No

PixelPercent

The percentage of image pixels that have a given dominant color.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Red

The Red RGB value for a dominant color.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

SimplifiedColor

One of 12 simplified color names applied to a dominant color.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Emotion

The emotions that appear to be expressed on the face, and the confidence level in the determination. The API is only making a determination of the physical appearance of a person's face. It is not a determination of the person's internal emotional state and should not be used in such a way. For example, a person pretending to have a sad face might not be sad emotionally.

Contents

Confidence

Level of confidence in the determination.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Type

Type of emotion detected.

Type: String

Valid Values: HAPPY | SAD | ANGRY | CONFUSED | DISGUSTED | SURPRISED | CALM | UNKNOWN | FEAR

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

EquipmentDetection

Information about an item of Personal Protective Equipment (PPE) detected by [DetectProtectiveEquipment \(p. 95\)](#). For more information, see [DetectProtectiveEquipment \(p. 95\)](#).

Contents

BoundingBox

A bounding box surrounding the item of detected PPE.

Type: [BoundingBox \(p. 259\)](#) object

Required: No

Confidence

The confidence that Amazon Rekognition has that the bounding box (BoundingBox) contains an item of PPE.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

CoversBodyPart

Information about the body part covered by the detected PPE.

Type: [CoversBodyPart \(p. 273\)](#) object

Required: No

Type

The type of detected PPE.

Type: String

Valid Values: FACE_COVER | HAND_COVER | HEAD_COVER

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

EvaluationResult

The evaluation results for the training of a model.

Contents

F1Score

The F1 score for the evaluation of all labels. The F1 score metric evaluates the overall precision and recall performance of the model as a single value. A higher value indicates better precision and recall performance. A lower score indicates that precision, recall, or both are performing poorly.

Type: Float

Required: No

Summary

The S3 bucket that contains the training summary.

Type: [Summary \(p. 374\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Eyeglasses

Indicates whether or not the face is wearing eye glasses, and the confidence level in the determination.

Contents

Confidence

Level of confidence in the determination.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Value

Boolean value that indicates whether the face is wearing eye glasses or not.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

EyeOpen

Indicates whether or not the eyes on the face are open, and the confidence level in the determination.

Contents

Confidence

Level of confidence in the determination.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Value

Boolean value that indicates whether the eyes on the face are open.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Face

Describes the face properties such as the bounding box, face ID, image ID of the input image, and external image ID that you assigned.

Contents

BoundingBox

Bounding box of the face.

Type: [BoundingBox \(p. 259\)](#) object

Required: No

Confidence

Confidence level that the bounding box contains a face (and not a different object such as a tree).

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

ExternalImageId

Identifier that you assign to all the faces in the input image.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\:]+

Required: No

FacetId

Unique identifier that Amazon Rekognition assigns to the face.

Type: String

Pattern: [0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}

Required: No

ImageId

Unique identifier that Amazon Rekognition assigns to the input image.

Type: String

Pattern: [0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}

Required: No

IndexFacesModelVersion

The version of the face detect and storage model that was used when indexing the face vector.

Type: String

Pattern: [0-9\.\.]+

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

FaceDetail

Structure containing attributes of the face that the algorithm detected.

A FaceDetail object contains either the default facial attributes or all facial attributes. The default attributes are BoundingBox, Confidence, Landmarks, Pose, and Quality.

[GetFaceDetection \(p. 118\)](#) is the only Amazon Rekognition Video stored video operation that can return a FaceDetail object with all attributes. To specify which attributes to return, use the FaceAttributes input parameter for [StartFaceDetection \(p. 205\)](#). The following Amazon Rekognition Video operations return only the default attributes. The corresponding Start operations don't have a FaceAttributes input parameter.

- [GetCelebrityRecognition](#)
- [GetPersonTracking](#)
- [GetFaceSearch](#)

The Amazon Rekognition Image [DetectFaces \(p. 79\)](#) and [IndexFaces \(p. 148\)](#) operations can return all facial attributes. To specify which attributes to return, use the Attributes input parameter for DetectFaces. For IndexFaces, use the DetectAttributes input parameter.

Contents

AgeRange

The estimated age range, in years, for the face. Low represents the lowest estimated age and High represents the highest estimated age.

Type: [AgeRange \(p. 254\)](#) object

Required: No

Beard

Indicates whether or not the face has a beard, and the confidence level in the determination.

Type: [Beard \(p. 257\)](#) object

Required: No

BoundingBox

Bounding box of the face. Default attribute.

Type: [BoundingBox \(p. 259\)](#) object

Required: No

Confidence

Confidence level that the bounding box contains a face (and not a different object such as a tree). Default attribute.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Emotions

The emotions that appear to be expressed on the face, and the confidence level in the determination. The API is only making a determination of the physical appearance of a person's face. It is not a determination of the person's internal emotional state and should not be used in such a way. For example, a person pretending to have a sad face might not be sad emotionally.

Type: Array of [Emotion \(p. 295\)](#) objects

Required: No

Eyeglasses

Indicates whether or not the face is wearing eye glasses, and the confidence level in the determination.

Type: [Eyeglasses \(p. 298\)](#) object

Required: No

EyesOpen

Indicates whether or not the eyes on the face are open, and the confidence level in the determination.

Type: [EyeOpen \(p. 299\)](#) object

Required: No

Gender

The predicted gender of a detected face.

Type: [Gender \(p. 309\)](#) object

Required: No

Landmarks

Indicates the location of landmarks on the face. Default attribute.

Type: Array of [Landmark \(p. 329\)](#) objects

Required: No

MouthOpen

Indicates whether or not the mouth on the face is open, and the confidence level in the determination.

Type: [MouthOpen \(p. 331\)](#) object

Required: No

Mustache

Indicates whether or not the face has a mustache, and the confidence level in the determination.

Type: [Mustache \(p. 332\)](#) object

Required: No

Pose

Indicates the pose of the face as determined by its pitch, roll, and yaw. Default attribute.

Type: [Pose \(p. 340\)](#) object

Required: No

Quality

Identifies image brightness and sharpness. Default attribute.

Type: [ImageQuality \(p. 318\)](#) object

Required: No

Smile

Indicates whether or not the face is smiling, and the confidence level in the determination.

Type: [Smile \(p. 360\)](#) object

Required: No

Sunglasses

Indicates whether or not the face is wearing sunglasses, and the confidence level in the determination.

Type: [Sunglasses \(p. 375\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

FaceDetection

Information about a face detected in a video analysis request and the time the face was detected in the video.

Contents

Face

The face properties for the detected face.

Type: [FaceDetail \(p. 302\)](#) object

Required: No

Timestamp

Time, in milliseconds from the start of the video, that the face was detected. Note that **Timestamp** is not guaranteed to be accurate to the individual frame where the face first appears.

Type: Long

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

FaceMatch

Provides face metadata. In addition, it also provides the confidence in the match of this face with the input face.

Contents

Face

Describes the face properties such as the bounding box, face ID, image ID of the source image, and external image ID that you assigned.

Type: [Face \(p. 300\)](#) object

Required: No

Similarity

Confidence in the match of this face with the input face.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

FaceRecord

Object containing both the face metadata (stored in the backend database), and facial attributes that are detected but aren't stored in the database.

Contents

Face

Describes the face properties such as the bounding box, face ID, image ID of the input image, and external image ID that you assigned.

Type: [Face \(p. 300\)](#) object

Required: No

FaceDetail

Structure containing attributes of the face that the algorithm detected.

Type: [FaceDetail \(p. 302\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

FaceSearchSettings

Input face recognition parameters for an Amazon Rekognition stream processor. Includes the collection to use for face recognition and the face attributes to detect. Defining the settings is required in the request parameter for [CreateStreamProcessor \(p. 31\)](#).

Contents

CollectionId

The ID of a collection that contains faces that you want to search for.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: No

FaceMatchThreshold

Minimum face match confidence score that must be met to return a result for a recognized face. The default is 80. 0 is the lowest confidence. 100 is the highest confidence. Values between 0 and 100 are accepted, and values lower than 80 are set to 80.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Gender

The predicted gender of a detected face.

Amazon Rekognition makes gender binary (male/female) predictions based on the physical appearance of a face in a particular image. This kind of prediction is not designed to categorize a person's gender identity, and you shouldn't use Amazon Rekognition to make such a determination. For example, a male actor wearing a long-haired wig and earrings for a role might be predicted as female.

Using Amazon Rekognition to make gender binary predictions is best suited for use cases where aggregate gender distribution statistics need to be analyzed without identifying specific users. For example, the percentage of female users compared to male users on a social media platform.

We don't recommend using gender binary predictions to make decisions that impact an individual's rights, privacy, or access to services.

Contents

Confidence

Level of confidence in the prediction.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Value

The predicted gender of the face.

Type: String

Valid Values: Male | Female

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

GeneralLabelsSettings

Contains filters for the object labels returned by DetectLabels. Filters can be inclusive, exclusive, or a combination of both and can be applied to individual labels or entire label categories.

Contents

LabelCategoryExclusionFilters

The label categories that should be excluded from the return from DetectLabels.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

Length Constraints: Minimum length of 0. Maximum length of 50.

Pattern: [A-Za-z0-9àâèçñó'-_() .]*

Required: No

LabelCategoryInclusionFilters

The label categories that should be included in the return from DetectLabels.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

Length Constraints: Minimum length of 0. Maximum length of 50.

Pattern: [A-Za-z0-9àâèçñó'-_() .]*

Required: No

LabelExclusionFilters

The labels that should be excluded from the return from DetectLabels.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

Length Constraints: Minimum length of 0. Maximum length of 50.

Pattern: [A-Za-z0-9àâèçñó'-_() .]*

Required: No

LabelInclusionFilters

The labels that should be included in the return from DetectLabels.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

Length Constraints: Minimum length of 0. Maximum length of 50.

Pattern: [A-Za-z0-9àâèçñó'-_() .]*

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Geometry

Information about where an object ([DetectCustomLabels \(p. 74\)](#)) or text ([DetectText \(p. 99\)](#)) is located on an image.

Contents

BoundingBox

An axis-aligned coarse representation of the detected item's location on the image.

Type: [BoundingBox \(p. 259\)](#) object

Required: No

Polygon

Within the bounding box, a fine-grained polygon around the detected item.

Type: Array of [Point \(p. 339\)](#) objects

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

GroundTruthManifest

The S3 bucket that contains an Amazon Sagemaker Ground Truth format manifest file.

Contents

S3Object

Provides the S3 bucket name and object name.

The region for the S3 bucket containing the S3 object must match the region you use for Amazon Rekognition operations.

For Amazon Rekognition to process an S3 object, the user must have permission to access the S3 object. For more information, see [How Amazon Rekognition works with IAM](#).

Type: [S3Object \(p. 354\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

HumanLoopActivationOutput

Shows the results of the human in the loop evaluation. If there is no HumanLoopArn, the input did not trigger human review.

Contents

HumanLoopActivationConditionsEvaluationResults

Shows the result of condition evaluations, including those conditions which activated a human review.

Type: String

Length Constraints: Maximum length of 10240.

Required: No

HumanLoopActivationReasons

Shows if and why human review was needed.

Type: Array of strings

Array Members: Minimum number of 1 item.

Required: No

HumanLoopArn

The Amazon Resource Name (ARN) of the HumanLoop created.

Type: String

Length Constraints: Maximum length of 256.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

HumanLoopConfig

Sets up the flow definition the image will be sent to if one of the conditions is met. You can also set certain attributes of the image before review.

Contents

DataAttributes

Sets attributes of the input data.

Type: [HumanLoopDataAttributes \(p. 316\)](#) object

Required: No

FlowDefinitionArn

The Amazon Resource Name (ARN) of the flow definition. You can create a flow definition by using the Amazon Sagemaker [CreateFlowDefinition](#) Operation.

Type: String

Length Constraints: Maximum length of 256.

Required: Yes

HumanLoopName

The name of the human review used for this image. This should be kept unique within a region.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 63.

Pattern: ^[a-zA-Z0-9](-*[a-zA-Z0-9])*

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

HumanLoopDataAttributes

Allows you to set attributes of the image. Currently, you can declare an image as free of personally identifiable information.

Contents

ContentClassifiers

Sets whether the input image is free of personally identifiable information.

Type: Array of strings

Array Members: Maximum number of 256 items.

Valid Values: `FreeOfPersonallyIdentifiableInformation` | `FreeOfAdultContent`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Image

Provides the input image either as bytes or an S3 object.

You pass image bytes to an Amazon Rekognition API operation by using the Bytes property. For example, you would use the Bytes property to pass an image loaded from a local file system. Image bytes passed by using the Bytes property must be base64-encoded. Your code may not need to encode image bytes if you are using an AWS SDK to call Amazon Rekognition API operations.

For more information, see [Analyzing an image loaded from a local file system](#) .

You pass images stored in an S3 bucket to an Amazon Rekognition API operation by using the S3Object property. Images stored in an S3 bucket do not need to be base64-encoded.

The region for the S3 bucket containing the S3 object must match the region you use for Amazon Rekognition operations.

If you use the AWS CLI to call Amazon Rekognition operations, passing image bytes using the Bytes property is not supported. You must first upload the image to an Amazon S3 bucket and then call the operation using the S3Object property.

For Amazon Rekognition to process an S3 object, the user must have permission to access the S3 object. For more information, see [How Amazon Rekognition works with IAM](#).

Contents

Bytes

Blob of image bytes up to 5 MBs.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 1. Maximum length of 5242880.

Required: No

S3Object

Identifies an S3 object as the image source.

Type: [S3Object \(p. 354\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ImageQuality

Identifies face image brightness and sharpness.

Contents

Brightness

Value representing brightness of the face. The service returns a value between 0 and 100 (inclusive). A higher value indicates a brighter face image.

Type: Float

Required: No

Sharpness

Value representing sharpness of the face. The service returns a value between 0 and 100 (inclusive). A higher value indicates a sharper face image.

Type: Float

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Instance

An instance of a label returned by Amazon Rekognition Image ([DetectLabels \(p. 84\)](#)) or by Amazon Rekognition Video ([GetLabelDetection \(p. 129\)](#)).

Contents

BoundingBox

The position of the label instance on the image.

Type: [BoundingBox \(p. 259\)](#) object

Required: No

Confidence

The confidence that Amazon Rekognition has in the accuracy of the bounding box.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

DominantColors

The dominant colors found in an individual instance of a label.

Type: Array of [DominantColor \(p. 293\)](#) objects

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

KinesisDataStream

The Kinesis data stream Amazon Rekognition to which the analysis results of a Amazon Rekognition stream processor are streamed. For more information, see [CreateStreamProcessor \(p. 31\)](#).

Contents

Arn

ARN of the output Amazon Kinesis Data Streams stream.

Type: String

Pattern: (^arn:([a-z\d-]+):kinesis:([a-z\d-]+):\d{12}:+\$)

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

KinesisVideoStream

Kinesis video stream stream that provides the source streaming video for a Amazon Rekognition Video stream processor. For more information, see [CreateStreamProcessor \(p. 31\)](#).

Contents

Arn

ARN of the Kinesis video stream stream that streams the source video.

Type: String

Pattern: (^arn:([a-z\d-]+):kinesisvideo:([a-z\d-]+):\d{12}:+\$)

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

KinesisVideoStreamStartSelector

Specifies the starting point in a Kinesis stream to start processing. You can use the producer timestamp or the fragment number. One of either producer timestamp or fragment number is required. If you use the producer timestamp, you must put the time in milliseconds. For more information about fragment numbers, see [Fragment](#).

Contents

FragmentNumber

The unique identifier of the fragment. This value monotonically increases based on the ingestion order.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: ^[0-9]+\$

Required: No

ProducerTimestamp

The timestamp from the producer corresponding to the fragment, in milliseconds, expressed in unix time format.

Type: Long

Valid Range: Minimum value of 0.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

KnownGender

The known gender identity for the celebrity that matches the provided ID. The known gender identity can be Male, Female, Nonbinary, or Unlisted.

Contents

Type

A string value of the KnownGender info about the Celebrity.

Type: String

Valid Values: Male | Female | Nonbinary | Unlisted

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Label

Structure containing details about the detected label, including the name, detected instances, parent labels, and level of confidence.

Contents

Aliases

A list of potential aliases for a given label.

Type: Array of [LabelAlias \(p. 326\)](#) objects

Required: No

Categories

A list of the categories associated with a given label.

Type: Array of [LabelCategory \(p. 327\)](#) objects

Required: No

Confidence

Level of confidence.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Instances

If Label represents an object, Instances contains the bounding boxes for each instance of the detected object. Bounding boxes are returned for common object labels such as people, cars, furniture, apparel or pets.

Type: Array of [Instance \(p. 319\)](#) objects

Required: No

Name

The name (label) of the object or scene.

Type: String

Required: No

Parents

The parent labels for a label. The response includes all ancestor labels.

Type: Array of [Parent \(p. 335\)](#) objects

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

LabelAlias

A potential alias of for a given label.

Contents

Name

The name of an alias for a given label.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

LabelCategory

The category that applies to a given label.

Contents

Name

The name of a category that applies to a given label.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

LabelDetection

Information about a label detected in a video analysis request and the time the label was detected in the video.

Contents

Label

Details about the detected label.

Type: [Label \(p. 324\)](#) object

Required: No

Timestamp

Time, in milliseconds from the start of the video, that the label was detected. Note that **Timestamp** is not guaranteed to be accurate to the individual frame where the label first appears.

Type: Long

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Landmark

Indicates the location of the landmark on the face.

Contents

Type

Type of landmark.

Type: String

Valid Values: eyeLeft | eyeRight | nose | mouthLeft | mouthRight | leftEyeBrowLeft | leftEyeBrowRight | leftEyeBrowUp | rightEyeBrowLeft | rightEyeBrowRight | rightEyeBrowUp | leftEyeLeft | leftEyeRight | leftEyeUp | leftEyeDown | rightEyeLeft | rightEyeRight | rightEyeUp | rightEyeDown | noseLeft | noseRight | mouthUp | mouthDown | leftPupil | rightPupil | upperJawlineLeft | midJawlineLeft | chinBottom | midJawlineRight | upperJawlineRight

Required: No

X

The x-coordinate of the landmark expressed as a ratio of the width of the image. The x-coordinate is measured from the left-side of the image. For example, if the image is 700 pixels wide and the x-coordinate of the landmark is at 350 pixels, this value is 0.5.

Type: Float

Required: No

Y

The y-coordinate of the landmark expressed as a ratio of the height of the image. The y-coordinate is measured from the top of the image. For example, if the image height is 200 pixels and the y-coordinate of the landmark is at 50 pixels, this value is 0.25.

Type: Float

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ModerationLabel

Provides information about a single type of inappropriate, unwanted, or offensive content found in an image or video. Each type of moderated content has a label within a hierarchical taxonomy. For a list of moderation labels in Amazon Rekognition, see [Using the image and video moderation APIs](#). For more information, see [Moderating content](#).

Contents

Confidence

Specifies the confidence that Amazon Rekognition has that the label has been correctly identified.

If you don't specify the `MinConfidence` parameter in the call to `DetectModerationLabels`, the operation returns labels with a confidence value greater than or equal to 50 percent.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Name

The label name for the type of unsafe content detected in the image.

Type: String

Required: No

ParentName

The name for the parent label. Labels at the top level of the hierarchy have the parent label "".

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

MouthOpen

Indicates whether or not the mouth on the face is open, and the confidence level in the determination.

Contents

Confidence

Level of confidence in the determination.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Value

Boolean value that indicates whether the mouth on the face is open or not.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Mustache

Indicates whether or not the face has a mustache, and the confidence level in the determination.

Contents

Confidence

Level of confidence in the determination.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Value

Boolean value that indicates whether the face has mustache or not.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

NotificationChannel

The Amazon Simple Notification Service topic to which Amazon Rekognition publishes the completion status of a video analysis operation. For more information, see [Calling Amazon Rekognition Video operations](#). Note that the Amazon SNS topic must have a topic name that begins with *AmazonRekognition* if you are using the `AmazonRekognitionServiceRole` permissions policy to access the topic. For more information, see [Giving access to multiple Amazon SNS topics](#).

Contents

RoleArn

The ARN of an IAM role that gives Amazon Rekognition publishing permissions to the Amazon SNS topic.

Type: String

Pattern: `arn:aws:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\\-/_]+`

Required: Yes

SNSTopicArn

The Amazon SNS topic to which Amazon Rekognition posts the completion status.

Type: String

Pattern: `(^arn:aws:sns:.*:\w{12}:\.+$)`

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

OutputConfig

The S3 bucket and folder location where training output is placed.

Contents

S3Bucket

The S3 bucket where training output is placed.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 255.

Pattern: [0-9A-Za-z\.\-_]*

Required: No

S3KeyPrefix

The prefix applied to the training output files.

Type: String

Length Constraints: Maximum length of 1024.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Parent

A parent label for a label. A label can have 0, 1, or more parents.

Contents

Name

The name of the parent label.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

PersonDetail

Details about a person detected in a video analysis request.

Contents

BoundingBox

Bounding box around the detected person.

Type: [BoundingBox \(p. 259\)](#) object

Required: No

Face

Face details for the detected person.

Type: [FaceDetail \(p. 302\)](#) object

Required: No

Index

Identifier for the person detected person within a video. Use to keep track of the person throughout the video. The identifier is not stored by Amazon Rekognition.

Type: Long

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

PersonDetection

Details and path tracking information for a single time a person's path is tracked in a video. Amazon Rekognition operations that track people's paths return an array of PersonDetection objects with elements for each time a person's path is tracked in a video.

For more information, see [GetPersonTracking \(p. 134\)](#).

Contents

Person

Details about a person whose path was tracked in a video.

Type: [PersonDetail \(p. 336\)](#) object

Required: No

Timestamp

The time, in milliseconds from the start of the video, that the person's path was tracked. Note that Timestamp is not guaranteed to be accurate to the individual frame where the person's path first appears.

Type: Long

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

PersonMatch

Information about a person whose face matches a face(s) in an Amazon Rekognition collection. Includes information about the faces in the Amazon Rekognition collection ([FaceMatch \(p. 306\)](#)), information about the person ([PersonDetail \(p. 336\)](#)), and the time stamp for when the person was detected in a video. An array of PersonMatch objects is returned by [GetFaceSearch \(p. 123\)](#).

Contents

FaceMatches

Information about the faces in the input collection that match the face of a person in the video.

Type: Array of [FaceMatch \(p. 306\)](#) objects

Required: No

Person

Information about the matched person.

Type: [PersonDetail \(p. 336\)](#) object

Required: No

Timestamp

The time, in milliseconds from the beginning of the video, that the person was matched in the video.

Type: Long

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Point

The X and Y coordinates of a point on an image or video frame. The X and Y values are ratios of the overall image size or video resolution. For example, if an input image is 700x200 and the values are X=0.5 and Y=0.25, then the point is at the (350,50) pixel coordinate on the image.

An array of Point objects, Polygon, is returned by [DetectText \(p. 99\)](#) and [DetectCustomLabels \(p. 74\)](#) or used to define regions of interest in Amazon Rekognition Video operations such as `CreateStreamProcessor`. Polygon represents a fine-grained polygon around a detected item. For more information, see [Geometry \(p. 312\)](#).

Contents

X

The value of the X coordinate for a point on a Polygon.

Type: Float

Required: No

Y

The value of the Y coordinate for a point on a Polygon.

Type: Float

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Pose

Indicates the pose of the face as determined by its pitch, roll, and yaw.

Contents

Pitch

Value representing the face rotation on the pitch axis.

Type: Float

Valid Range: Minimum value of -180. Maximum value of 180.

Required: No

Roll

Value representing the face rotation on the roll axis.

Type: Float

Valid Range: Minimum value of -180. Maximum value of 180.

Required: No

Yaw

Value representing the face rotation on the yaw axis.

Type: Float

Valid Range: Minimum value of -180. Maximum value of 180.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ProjectDescription

A description of an Amazon Rekognition Custom Labels project. For more information, see [DescribeProjects \(p. 61\)](#).

Contents

CreationTimestamp

The Unix timestamp for the date and time that the project was created.

Type: [Timestamp](#)

Required: No

Datasets

Information about the training and test datasets in the project.

Type: Array of [DatasetMetadata \(p. 280\)](#) objects

Required: No

ProjectArn

The Amazon Resource Name (ARN) of the project.

Type: [String](#)

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\:\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\/[0-9]+\\$)

Required: No

Status

The current status of the project.

Type: [String](#)

Valid Values: CREATING | CREATED | DELETING

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ProjectPolicy

Describes a project policy in the response from [ListProjectPolicies \(p. 170\)](#).

Contents

CreationTimestamp

The Unix datetime for the creation of the project policy.

Type: Timestamp

Required: No

LastUpdatedTimestamp

The Unix datetime for when the project policy was last updated.

Type: Timestamp

Required: No

PolicyDocument

The JSON document for the project policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2000.

Pattern: [\u0009\u000A\u000D\u0020-\u00FF]+

Required: No

PolicyName

The name of the project policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: [a-zA-Z0-9_.\u0020-]+

Required: No

PolicyRevisionId

The revision ID of the project policy.

Type: String

Length Constraints: Maximum length of 64.

Pattern: [0-9A-Fa-f]+

Required: No

ProjectArn

The Amazon Resource Name (ARN) of the project to which the project policy is attached.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+:\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\/[0-9]+\\$)

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ProjectVersionDescription

A description of a version of an Amazon Rekognition Custom Labels model.

Contents

BillableTrainingTimeInSeconds

The duration, in seconds, that you were billed for a successful training of the model version. This value is only returned if the model version has been successfully trained.

Type: Long

Valid Range: Minimum value of 0.

Required: No

CreationTimestamp

The Unix datetime for the date and time that training started.

Type: Timestamp

Required: No

EvaluationResult

The training results. EvaluationResult is only returned if training is successful.

Type: [EvaluationResult \(p. 297\)](#) object

Required: No

KmsKeyId

The identifier for the AWS Key Management Service key (AWS KMS key) that was used to encrypt the model during training.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: ^[A-Za-z0-9][A-Za-z0-9:_/+=,@.-]{0,2048}\$

Required: No

ManifestSummary

The location of the summary manifest. The summary manifest provides aggregate data validation results for the training and test datasets.

Type: [GroundTruthManifest \(p. 313\)](#) object

Required: No

MaxInferenceUnits

The maximum number of inference units Amazon Rekognition Custom Labels uses to auto-scale the model. For more information, see [StartProjectVersion \(p. 221\)](#).

Type: Integer

Valid Range: Minimum value of 1.

Required: No

MinInferenceUnits

The minimum number of inference units used by the model. For more information, see [StartProjectVersion \(p. 221\)](#).

Type: Integer

Valid Range: Minimum value of 1.

Required: No

OutputConfig

The location where training results are saved.

Type: [OutputConfig \(p. 334\)](#) object

Required: No

ProjectVersionArn

The Amazon Resource Name (ARN) of the model version.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\version\/[a-zA-Z0-9_.\-\-]{1,255}\[0-9]+\\$)

Required: No

SourceProjectVersionArn

If the model version was copied from a different project, SourceProjectVersionArn contains the ARN of the source model version.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: (^arn:[a-z\d-]+:rekognition:[a-z\d-]+\d{12}:project\/[a-zA-Z0-9_.\-\-]{1,255}\version\/[a-zA-Z0-9_.\-\-]{1,255}\[0-9]+\\$)

Required: No

Status

The current status of the model version.

Type: String

Valid Values: TRAINING_IN_PROGRESS | TRAINING_COMPLETED | TRAINING_FAILED | STARTING | RUNNING | FAILED | STOPPING | STOPPED | DELETING | COPYING_IN_PROGRESS | COPYING_COMPLETED | COPYING_FAILED

Required: No

StatusMessage

A descriptive message for an error or warning that occurred.

Type: String

Required: No

TestingDataResult

Contains information about the testing results.

Type: [TestingDataResult \(p. 378\)](#) object

Required: No

TrainingDataResult

Contains information about the training results.

Type: [TrainingDataResult \(p. 383\)](#) object

Required: No

TrainingEndTimestamp

The Unix date and time that training of the model ended.

Type: [Timestamp](#)

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ProtectiveEquipmentBodyPart

Information about a body part detected by [DetectProtectiveEquipment \(p. 95\)](#) that contains PPE. An array of ProtectiveEquipmentBodyPart objects is returned for each person detected by DetectProtectiveEquipment.

Contents

Confidence

The confidence that Amazon Rekognition has in the detection accuracy of the detected body part.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

EquipmentDetections

An array of Personal Protective Equipment items detected around a body part.

Type: Array of [EquipmentDetection \(p. 296\)](#) objects

Required: No

Name

The detected body part.

Type: String

Valid Values: FACE | HEAD | LEFT_HAND | RIGHT_HAND

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ProtectiveEquipmentPerson

A person detected by a call to [DetectProtectiveEquipment \(p. 95\)](#). The API returns all persons detected in the input image in an array of ProtectiveEquipmentPerson objects.

Contents

BodyParts

An array of body parts detected on a person's body (including body parts without PPE).

Type: Array of [ProtectiveEquipmentBodyPart \(p. 347\)](#) objects

Required: No

BoundingBox

A bounding box around the detected person.

Type: [BoundingBox \(p. 259\)](#) object

Required: No

Confidence

The confidence that Amazon Rekognition has that the bounding box contains a person.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Id

The identifier for the detected person. The identifier is only unique for a single call to [DetectProtectiveEquipment](#).

Type: Integer

Valid Range: Minimum value of 0.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ProtectiveEquipmentSummarizationAttributes

Specifies summary attributes to return from a call to [DetectProtectiveEquipment \(p. 95\)](#). You can specify which types of PPE to summarize. You can also specify a minimum confidence value for detections. Summary information is returned in the [Summary \(ProtectiveEquipmentSummary \(p. 350\)\)](#) field of the response from `DetectProtectiveEquipment`. The summary includes which persons in an image were detected wearing the requested types of person protective equipment (PPE), which persons were detected as not wearing PPE, and the persons in which a determination could not be made. For more information, see [ProtectiveEquipmentSummary \(p. 350\)](#).

Contents

MinConfidence

The minimum confidence level for which you want summary information. The confidence level applies to person detection, body part detection, equipment detection, and body part coverage. Amazon Rekognition doesn't return summary information with a confidence than this specified value. There isn't a default value.

Specify a `MinConfidence` value that is between 50-100% as `DetectProtectiveEquipment` returns predictions only where the detection confidence is between 50% - 100%. If you specify a value that is less than 50%, the results are the same specifying a value of 50%.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: Yes

RequiredEquipmentTypes

An array of personal protective equipment types for which you want summary information. If a person is detected wearing a required equipment type, the person's ID is added to the `PersonsWithRequiredEquipment` array field returned in [ProtectiveEquipmentSummary \(p. 350\)](#) by `DetectProtectiveEquipment`.

Type: Array of strings

Valid Values: FACE_COVER | HAND_COVER | HEAD_COVER

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ProtectiveEquipmentSummary

Summary information for required items of personal protective equipment (PPE) detected on persons by a call to [DetectProtectiveEquipment \(p. 95\)](#). You specify the required type of PPE in the `SummarizationAttributes` ([ProtectiveEquipmentSummarizationAttributes \(p. 349\)](#)) input parameter. The summary includes which persons were detected wearing the required personal protective equipment (`PersonsWithRequiredEquipment`), which persons were detected as not wearing the required PPE (`PersonsWithoutRequiredEquipment`), and the persons in which a determination could not be made (`PersonsIndeterminate`).

To get a total for each category, use the size of the field array. For example, to find out how many people were detected as wearing the specified PPE, use the size of the `PersonsWithRequiredEquipment` array. If you want to find out more about a person, such as the location ([BoundingBox \(p. 259\)](#)) of the person on the image, use the person ID in each array element. Each person ID matches the ID field of a [ProtectiveEquipmentPerson \(p. 348\)](#) object returned in the `Persons` array by `DetectProtectiveEquipment`.

Contents

PersonsIndeterminate

An array of IDs for persons where it was not possible to determine if they are wearing personal protective equipment.

Type: Array of integers

Valid Range: Minimum value of 0.

Required: No

PersonsWithoutRequiredEquipment

An array of IDs for persons who are not wearing all of the types of PPE specified in the `RequiredEquipmentTypes` field of the detected personal protective equipment.

Type: Array of integers

Valid Range: Minimum value of 0.

Required: No

PersonsWithRequiredEquipment

An array of IDs for persons who are wearing detected personal protective equipment.

Type: Array of integers

Valid Range: Minimum value of 0.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

RegionOfInterest

Specifies a location within the frame that Rekognition checks for objects of interest such as text, labels, or faces. It uses a BoundingBox or Polygon to set a region of the screen.

A word, face, or label is included in the region if it is more than half in that region. If there is more than one region, the word, face, or label is compared with all regions of the screen. Any object of interest that is more than half in a region is kept in the results.

Contents

BoundingBox

The box representing a region of interest on screen.

Type: [BoundingBox \(p. 259\)](#) object

Required: No

Polygon

Specifies a shape made up of up to 10 Point objects to define a region of interest.

Type: Array of [Point \(p. 339\)](#) objects

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

S3Destination

The Amazon S3 bucket location to which Amazon Rekognition publishes the detailed inference results of a video analysis operation. These results include the name of the stream processor resource, the session ID of the stream processing session, and labeled timestamps and bounding boxes for detected labels.

Contents

Bucket

The name of the Amazon S3 bucket you want to associate with the streaming video project. You must be the owner of the Amazon S3 bucket.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 255.

Pattern: [0-9A-Za-z\.\-_]*

Required: No

KeyPrefix

The prefix value of the location within the bucket that you want the information to be published to. For more information, see [Using prefixes](#).

Type: String

Length Constraints: Maximum length of 1024.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

S3Object

Provides the S3 bucket name and object name.

The region for the S3 bucket containing the S3 object must match the region you use for Amazon Rekognition operations.

For Amazon Rekognition to process an S3 object, the user must have permission to access the S3 object. For more information, see [How Amazon Rekognition works with IAM](#).

Contents

Bucket

Name of the S3 bucket.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 255.

Pattern: [0-9A-Za-z\.\-_]*

Required: No

Name

S3 object key name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Required: No

Version

If the bucket is versioning enabled, you can specify the object version.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

SegmentDetection

A technical cue or shot detection segment detected in a video. An array of SegmentDetection objects containing all segments detected in a stored video is returned by [GetSegmentDetection \(p. 139\)](#).

Contents

DurationFrames

The duration of a video segment, expressed in frames.

Type: Long

Valid Range: Minimum value of 0.

Required: No

DurationMillis

The duration of the detected segment in milliseconds.

Type: Long

Valid Range: Minimum value of 0.

Required: No

DurationSMPTE

The duration of the timecode for the detected segment in SMPTE format.

Type: String

Required: No

EndFrameNumber

The frame number at the end of a video segment, using a frame index that starts with 0.

Type: Long

Valid Range: Minimum value of 0.

Required: No

EndTimecodeSMPTE

The frame-accurate SMPTE timecode, from the start of a video, for the end of a detected segment. EndTimecode is in *HH:MM:SS:fr* format (and *;fr* for drop frame-rates).

Type: String

Required: No

EndTimestampMillis

The end time of the detected segment, in milliseconds, from the start of the video. This value is rounded down.

Type: Long

Required: No

ShotSegment

If the segment is a shot detection, contains information about the shot detection.

Type: [ShotSegment \(p. 359\)](#) object

Required: No

StartFrameNumber

The frame number of the start of a video segment, using a frame index that starts with 0.

Type: Long

Valid Range: Minimum value of 0.

Required: No

StartTimecodeSMPT

The frame-accurate SMPTE timecode, from the start of a video, for the start of a detected segment. StartTimecode is in *HH:MM:SS:fr* format (and *;fr* for drop frame-rates).

Type: String

Required: No

StartTimestampMillis

The start time of the detected segment in milliseconds from the start of the video. This value is rounded down. For example, if the actual timestamp is 100.6667 milliseconds, Amazon Rekognition Video returns a value of 100 millis.

Type: Long

Required: No

TechnicalCueSegment

If the segment is a technical cue, contains information about the technical cue.

Type: [TechnicalCueSegment \(p. 376\)](#) object

Required: No

Type

The type of the segment. Valid values are TECHNICAL_CUE and SHOT.

Type: String

Valid Values: TECHNICAL_CUE | SHOT

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for Ruby V3](#)

SegmentTypeInfo

Information about the type of a segment requested in a call to [StartSegmentDetection \(p. 224\)](#). An array of SegmentTypeInfo objects is returned by the response from [GetSegmentDetection \(p. 139\)](#).

Contents

ModelVersion

The version of the model used to detect segments.

Type: String

Required: No

Type

The type of a segment (technical cue or shot detection).

Type: String

Valid Values: TECHNICAL_CUE | SHOT

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ShotSegment

Information about a shot detection segment detected in a video. For more information, see [SegmentDetection \(p. 355\)](#).

Contents

Confidence

The confidence that Amazon Rekognition Video has in the accuracy of the detected segment.

Type: Float

Valid Range: Minimum value of 50. Maximum value of 100.

Required: No

Index

An Identifier for a shot detection segment detected in a video.

Type: Long

Valid Range: Minimum value of 0.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Smile

Indicates whether or not the face is smiling, and the confidence level in the determination.

Contents

Confidence

Level of confidence in the determination.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Value

Boolean value that indicates whether the face is smiling or not.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StartSegmentDetectionFilters

Filters applied to the technical cue or shot detection segments. For more information, see [StartSegmentDetection \(p. 224\)](#).

Contents

ShotFilter

Filters that are specific to shot detections.

Type: [StartShotDetectionFilter \(p. 362\)](#) object

Required: No

TechnicalCueFilter

Filters that are specific to technical cues.

Type: [StartTechnicalCueDetectionFilter \(p. 363\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StartShotDetectionFilter

Filters for the shot detection segments returned by [GetSegmentDetection](#). For more information, see [StartSegmentDetectionFilters \(p. 361\)](#).

Contents

MinSegmentConfidence

Specifies the minimum confidence that Amazon Rekognition Video must have in order to return a detected segment. Confidence represents how certain Amazon Rekognition is that a segment is correctly identified. 0 is the lowest confidence. 100 is the highest confidence. Amazon Rekognition Video doesn't return any segments with a confidence level lower than this specified value.

If you don't specify `MinSegmentConfidence`, the `GetSegmentDetection` returns segments with confidence values greater than or equal to 50 percent.

Type: `Float`

Valid Range: Minimum value of 50. Maximum value of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StartTechnicalCueDetectionFilter

Filters for the technical segments returned by [GetSegmentDetection \(p. 139\)](#). For more information, see [StartSegmentDetectionFilters \(p. 361\)](#).

Contents

BlackFrame

A filter that allows you to control the black frame detection by specifying the black levels and pixel coverage of black pixels in a frame. Videos can come from multiple sources, formats, and time periods, with different standards and varying noise levels for black frames that need to be accounted for.

Type: [BlackFrame \(p. 258\)](#) object

Required: No

MinSegmentConfidence

Specifies the minimum confidence that Amazon Rekognition Video must have in order to return a detected segment. Confidence represents how certain Amazon Rekognition is that a segment is correctly identified. 0 is the lowest confidence. 100 is the highest confidence. Amazon Rekognition Video doesn't return any segments with a confidence level lower than this specified value.

If you don't specify MinSegmentConfidence, GetSegmentDetection returns segments with confidence values greater than or equal to 50 percent.

Type: Float

Valid Range: Minimum value of 50. Maximum value of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StartTextDetectionFilters

Set of optional parameters that let you set the criteria text must meet to be included in your response. WordFilter looks at a word's height, width and minimum confidence. RegionOfInterest lets you set a specific region of the screen to look for text in.

Contents

RegionsOfInterest

Filter focusing on a certain area of the frame. Uses a BoundingBox object to set the region of the screen.

Type: Array of [RegionOfInterest \(p. 352\)](#) objects

Array Members: Minimum number of 0 items. Maximum number of 10 items.

Required: No

WordFilter

Filters focusing on qualities of the text, such as confidence or size.

Type: [DetectionFilter \(p. 284\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StreamProcessingStartSelector

This is a required parameter for label detection stream processors and should not be used to start a face search stream processor.

Contents

KVSStreamStartSelector

Specifies the starting point in the stream to start processing. This can be done with a producer timestamp or a fragment number in a Kinesis stream.

Type: [KinesisVideoStreamStartSelector \(p. 322\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StreamProcessingStopSelector

Specifies when to stop processing the stream. You can specify a maximum amount of time to process the video.

Contents

MaxDurationInSeconds

Specifies the maximum amount of time in seconds that you want the stream to be processed. The largest amount of time is 2 minutes. The default is 10 seconds.

Type: Long

Valid Range: Minimum value of 1. Maximum value of 120.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StreamProcessor

An object that recognizes faces or labels in a streaming video. An Amazon Rekognition stream processor is created by a call to [CreateStreamProcessor \(p. 31\)](#). The request parameters for `CreateStreamProcessor` describe the Kinesis video stream source for the streaming video, face recognition parameters, and where to stream the analysis results.

Contents

Name

Name of the Amazon Rekognition stream processor.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: [a-zA-Z0-9_.\-\-]+

Required: No

Status

Current status of the Amazon Rekognition stream processor.

Type: String

Valid Values: STOPPED | STARTING | RUNNING | FAILED | STOPPING | UPDATING

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StreamProcessorDataSharingPreference

Allows you to opt in or opt out to share data with Rekognition to improve model performance. You can choose this option at the account level or on a per-stream basis. Note that if you opt out at the account level this setting is ignored on individual streams.

Contents

OptIn

If this option is set to true, you choose to share data with Rekognition to improve model performance.

Type: Boolean

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StreamProcessorInput

Information about the source streaming video.

Contents

KinesisVideoStream

The Kinesis video stream input stream for the source streaming video.

Type: [KinesisVideoStream \(p. 321\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StreamProcessorNotificationChannel

The Amazon Simple Notification Service topic to which Amazon Rekognition publishes the object detection results and completion status of a video analysis operation.

Amazon Rekognition publishes a notification the first time an object of interest or a person is detected in the video stream. For example, if Amazon Rekognition detects a person at second 2, a pet at second 4, and a person again at second 5, Amazon Rekognition sends 2 object class detected notifications, one for a person at second 2 and one for a pet at second 4.

Amazon Rekognition also publishes an end-of-session notification with a summary when the stream processing session is complete.

Contents

SNSTopicArn

The Amazon Resource Number (ARN) of the Amazon Simple Notification Service topic to which Amazon Rekognition posts the completion status.

Type: String

Pattern: (^arn:aws:sns:.*:\w{12}:\.+\$)

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StreamProcessorOutput

Information about the Amazon Kinesis Data Streams stream to which a Amazon Rekognition Video stream processor streams the results of a video analysis. For more information, see [CreateStreamProcessor \(p. 31\)](#).

Contents

KinesisDataStream

The Amazon Kinesis Data Streams stream to which the Amazon Rekognition stream processor streams the analysis results.

Type: [KinesisDataStream \(p. 320\)](#) object

Required: No

S3Destination

The Amazon S3 bucket location to which Amazon Rekognition publishes the detailed inference results of a video analysis operation.

Type: [S3Destination \(p. 353\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StreamProcessorSettings

Input parameters used in a streaming video analyzed by a Amazon Rekognition stream processor. You can use FaceSearch to recognize faces in a streaming video, or you can use ConnectedHome to detect labels.

Contents

ConnectedHome

Label detection settings to use on a streaming video. Defining the settings is required in the request parameter for [CreateStreamProcessor \(p. 31\)](#). Including this setting in the `CreateStreamProcessor` request enables you to use the stream processor for label detection. You can then select what you want the stream processor to detect, such as people or pets. When the stream processor has started, one notification is sent for each object class specified. For example, if packages and pets are selected, one SNS notification is published the first time a package is detected and one SNS notification is published the first time a pet is detected, as well as an end-of-session summary.

Type: [ConnectedHomeSettings \(p. 270\)](#) object

Required: No

FaceSearch

Face search settings to use on a streaming video.

Type: [FaceSearchSettings \(p. 308\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

StreamProcessorSettingsForUpdate

The stream processor settings that you want to update. ConnectedHome settings can be updated to detect different labels with a different minimum confidence.

Contents

ConnectedHomeForUpdate

The label detection settings you want to use for your stream processor.

Type: [ConnectedHomeSettingsForUpdate \(p. 271\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Summary

The S3 bucket that contains the training summary. The training summary includes aggregated evaluation metrics for the entire testing dataset and metrics for each individual label.

You get the training summary S3 bucket location by calling [DescribeProjectVersions \(p. 64\)](#).

Contents

S3Object

Provides the S3 bucket name and object name.

The region for the S3 bucket containing the S3 object must match the region you use for Amazon Rekognition operations.

For Amazon Rekognition to process an S3 object, the user must have permission to access the S3 object. For more information, see [How Amazon Rekognition works with IAM](#).

Type: [S3Object \(p. 354\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Sunglasses

Indicates whether or not the face is wearing sunglasses, and the confidence level in the determination.

Contents

Confidence

Level of confidence in the determination.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

Value

Boolean value that indicates whether the face is wearing sunglasses or not.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

TechnicalCueSegment

Information about a technical cue segment. For more information, see [SegmentDetection \(p. 355\)](#).

Contents

Confidence

The confidence that Amazon Rekognition Video has in the accuracy of the detected segment.

Type: Float

Valid Range: Minimum value of 50. Maximum value of 100.

Required: No

Type

The type of the technical cue.

Type: String

Valid Values: ColorBars | EndCredits | BlackFrames | OpeningCredits | StudioLogo | Slate | Content

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

TestingData

The dataset used for testing. Optionally, if AutoCreate is set, Amazon Rekognition Custom Labels uses the training dataset to create a test dataset with a temporary split of the training dataset.

Contents

Assets

The assets used for testing.

Type: Array of [Asset \(p. 255\)](#) objects

Required: No

AutoCreate

If specified, Amazon Rekognition Custom Labels temporarily splits the training dataset (80%) to create a test dataset (20%) for the training job. After training completes, the test dataset is not stored and the training dataset reverts to its previous size.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

TestingDataResult

Sagemaker Groundtruth format manifest files for the input, output and validation datasets that are used and created during testing.

Contents

Input

The testing dataset that was supplied for training.

Type: [TestingData \(p. 377\)](#) object

Required: No

Output

The subset of the dataset that was actually tested. Some images (assets) might not be tested due to file formatting and other issues.

Type: [TestingData \(p. 377\)](#) object

Required: No

Validation

The location of the data validation manifest. The data validation manifest is created for the test dataset during model training.

Type: [ValidationData \(p. 385\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

TextDetection

Information about a word or line of text detected by [DetectText \(p. 99\)](#).

The **DetectedText** field contains the text that Amazon Rekognition detected in the image.

Every word and line has an identifier (**Id**). Each word belongs to a line and has a parent identifier (**ParentId**) that identifies the line of text in which the word appears. The word **Id** is also an index for the word within a line of words.

For more information, see [Detecting text](#).

Contents

Confidence

The confidence that Amazon Rekognition has in the accuracy of the detected text and the accuracy of the geometry points around the detected text.

Type: Float

Valid Range: Minimum value of 0. Maximum value of 100.

Required: No

DetectedText

The word or line of text recognized by Amazon Rekognition.

Type: String

Required: No

Geometry

The location of the detected text on the image. Includes an axis aligned coarse bounding box surrounding the text and a finer grain polygon for more accurate spatial information.

Type: [Geometry \(p. 312\)](#) object

Required: No

Id

The identifier for the detected text. The identifier is only unique for a single call to [DetectText](#).

Type: Integer

Valid Range: Minimum value of 0.

Required: No

ParentId

The Parent identifier for the detected text identified by the value of ID. If the type of detected text is LINE, the value of ParentId is Null.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

Type

The type of text that was detected.

Type: String

Valid Values: LINE | WORD

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

TextDetectionResult

Information about text detected in a video. Includes the detected text, the time in milliseconds from the start of the video that the text was detected, and where it was detected on the screen.

Contents

TextDetection

Details about text detected in a video.

Type: [TextDetection \(p. 379\)](#) object

Required: No

Timestamp

The time, in milliseconds from the start of the video, that the text was detected. Note that Timestamp is not guaranteed to be accurate to the individual frame where the text first appears.

Type: Long

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

TrainingData

The dataset used for training.

Contents

Assets

A Sagemaker GroundTruth manifest file that contains the training images (assets).

Type: Array of [Asset \(p. 255\)](#) objects

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

TrainingDataResult

Sagemaker Groundtruth format manifest files for the input, output and validation datasets that are used and created during testing.

Contents

Input

The training assets that you supplied for training.

Type: [TrainingData \(p. 382\)](#) object

Required: No

Output

The images (assets) that were actually trained by Amazon Rekognition Custom Labels.

Type: [TrainingData \(p. 382\)](#) object

Required: No

Validation

The location of the data validation manifest. The data validation manifest is created for the training dataset during model training.

Type: [ValidationData \(p. 385\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

UnindexedFace

A face that [IndexFaces \(p. 148\)](#) detected, but didn't index. Use the Reasons response attribute to determine why a face wasn't indexed.

Contents

FaceDetail

The structure that contains attributes of a face that IndexFacesdetected, but didn't index.

Type: [FaceDetail \(p. 302\)](#) object

Required: No

Reasons

An array of reasons that specify why a face wasn't indexed.

- EXTREME_POSE - The face is at a pose that can't be detected. For example, the head is turned too far away from the camera.
- EXCEEDS_MAX_FACES - The number of faces detected is already higher than that specified by the MaxFaces input parameter for IndexFaces.
- LOW_BRIGHTNESS - The image is too dark.
- LOW_SHARPNESS - The image is too blurry.
- LOW_CONFIDENCE - The face was detected with a low confidence.
- SMALL_BOUNDING_BOX - The bounding box around the face is too small.

Type: Array of strings

Valid Values: EXCEEDS_MAX_FACES | EXTREME_POSE | LOW_BRIGHTNESS | LOW_SHARPNESS | LOW_CONFIDENCE | SMALL_BOUNDING_BOX | LOW_FACE_QUALITY

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ValidationData

Contains the Amazon S3 bucket location of the validation data for a model training job.

The validation data includes error information for individual JSON lines in the dataset. For more information, see [Debugging a Failed Model Training](#).

You get the ValidationData object for the training dataset ([TrainingDataResult \(p. 383\)](#)) and the test dataset ([TestingDataResult \(p. 378\)](#)) by calling [DescribeProjectVersions \(p. 64\)](#).

The assets array contains a single [Asset \(p. 255\)](#) object. The [GroundTruthManifest \(p. 313\)](#) field of the Asset object contains the S3 bucket location of the validation data.

Contents

Assets

The assets that comprise the validation data.

Type: Array of [Asset \(p. 255\)](#) objects

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Video

Video file stored in an Amazon S3 bucket. Amazon Rekognition video start operations such as [StartLabelDetection \(p. 213\)](#) use Video to specify a video for analysis. The supported file formats are .mp4, .mov and .avi.

Contents

S3Object

The Amazon S3 bucket name and file name for the video.

Type: [S3Object \(p. 354\)](#) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

VideoMetadata

Information about a video that Amazon Rekognition analyzed. Videodata is returned in every page of paginated responses from a Amazon Rekognition video operation.

Contents

Codec

Type of compression used in the analyzed video.

Type: String

Required: No

ColorRange

A description of the range of luminance values in a video, either LIMITED (16 to 235) or FULL (0 to 255).

Type: String

Valid Values: FULL | LIMITED

Required: No

DurationMillis

Length of the video in milliseconds.

Type: Long

Valid Range: Minimum value of 0.

Required: No

Format

Format of the analyzed video. Possible values are MP4, MOV and AVI.

Type: String

Required: No

FrameHeight

Vertical pixel dimension of the video.

Type: Long

Valid Range: Minimum value of 0.

Required: No

FrameRate

Number of frames per second in the video.

Type: Float

Required: No

FrameWidth

Horizontal pixel dimension of the video.

Type: Long

Valid Range: Minimum value of 0.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see [Signature Version 4 Signing Process](#) in the *Amazon Web Services General Reference*.

Action

The action to be performed.

Type: string

Required: Yes

Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

X-Amz-Algorithm

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: AWS4-HMAC-SHA256

Required: Conditional

X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: *access_key/YYYYMMDD/region/service/aws4_request*.

For more information, see [Task 2: Create a String to Sign for Signature Version 4](#) in the *Amazon Web Services General Reference*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-Date

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: 20120325T120000Z.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is

not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see [Handling Dates in Signature Version 4](#) in the *Amazon Web Services General Reference*.

Type: string

Required: Conditional

X-Amz-Security-Token

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS Security Token Service, go to [AWS Services That Work with IAM](#) in the *IAM User Guide*.

Condition: If you're using temporary security credentials from the AWS Security Token Service, you must include the security token.

Type: string

Required: Conditional

X-Amz-Signature

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see [Task 1: Create a Canonical Request For Signature Version 4](#) in the *Amazon Web Services General Reference*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

InternalFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

InvalidAction

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

InvalidClientId

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

InvalidParameterCombination

Parameters that must not be used together were used together.

HTTP Status Code: 400

InvalidParameterValue

An invalid or out-of-range value was supplied for the input parameter.

HTTP Status Code: 400

InvalidQueryParameter

The AWS query string is malformed or does not adhere to AWS standards.

HTTP Status Code: 400

MalformedQueryString

The query string contains a syntax error.

HTTP Status Code: 404

MissingAction

The request is missing an action or a required parameter.

HTTP Status Code: 400

MissingAuthenticationToken

The request must contain either a valid (registered) AWS access key ID or X.509 certificate.

HTTP Status Code: 403

MissingParameter

A required parameter for the specified action is not supplied.

HTTP Status Code: 400

NotAuthorized

You do not have permission to perform this action.

HTTP Status Code: 400

OptInRequired

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

RequestExpired

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationException

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400