Cloud Vision API

Integrates Google Vision features, including image labeling, face, logo, and landmark detection, optical character recognition (OCR), and detection of explicit content, into applications.

Service: vision.googleapis.com

To call this service, we recommend that you use the Google-provided client libraries (https://cloud.google.com/apis/docs/client-libraries-explained). If your application needs to use your own libraries to call this service, use the following information when you make the API requests.

Discovery document

A Discovery Document (https://developers.google.com/discovery/v1/reference/apis) is a machine-readable specification for describing and consuming REST APIs. It is used to build client libraries, IDE plugins, and other tools that interact with Google APIs. One service may provide multiple discovery documents. This service provides the following discovery document:

- https://vision.googleapis.com/$discovery/rest?version=v1 (https://vision.googleapis.com/$discovery/rest?version=v1)

Service endpoint

A service endpoint (https://cloud.google.com/apis/design/glossary#api_service_endpoint) is a base URL that specifies the network address of an API service. One service might have multiple service endpoints. This service has the following service endpoint and all URIs below are relative to this service endpoint:

- `https://vision.googleapis.com`

REST Resource: v1.files (/vision/docs/reference/rest/v1/files)

Methods	
`annotate` (/vision/docs/reference/rest/v1/files/annotate)	`POST /v1/files:annotate` Service that performs image detection and annotation for a batch of files.
`asyncBatchAnnotate` (/vision/docs/reference/rest/v1/files/asyncBatchAnnotate)	`POST /v1/files:asyncBatchAnnotate` Run asynchronous image detection and annotation for a list of generic files, such as PDF files, which may contain multiple pages and multiple images per page.

REST Resource: v1.images (/vision/docs/reference/rest/v1/images)

Methods	
`annotate` (/vision/docs/reference/rest/v1/images/annotate)	`POST /v1/images:annotate` Run image detection and annotation for a batch of images.
`asyncBatchAnnotate` (/vision/docs/reference/rest/v1/images/asyncBatchAnnotate)	`POST /v1/images:asyncBatchAnnotate` Run asynchronous image detection and annotation for a list of images.

REST Resource: v1.locations.operations

(/vision/docs/reference/rest/v1/locations.operations)

Methods	
`get` (/vision/docs/reference/rest/v1/locations.operations/get)	`GET /v1/{name=locations/*/operations/*}` Gets the latest state of a long-running operation.

REST Resource: v1.operations (/vision/docs/reference/rest/v1/operations)

Methods	
`cancel` (/vision/docs/reference/rest/v1/operations/cancel)	**`POST /v1/{name=operations/**}:cancel`** Starts asynchronous cancellation on a long-running operation.
`delete` (/vision/docs/reference/rest/v1/operations/delete)	**`DELETE /v1/{name=operations/**}`** Deletes a long-running operation.
`get` (/vision/docs/reference/rest/v1/operations/get)	**`GET /v1/{name=operations/*}`** Gets the latest state of a long-running operation.
`list` (/vision/docs/reference/rest/v1/operations/list)	**`GET /v1/{name}`** Lists operations that match the specified filter in the request.

REST Resource: v1.projects.files (/vision/docs/reference/rest/v1/projects.files)

Methods	
`annotate` (/vision/docs/reference/rest/v1/projects.files/annotate)	**`POST /v1/{parent=projects/*}/files:annotate`** Service that performs image detection and annotation for a batch of files.
`asyncBatchAnnotate` (/vision/docs/reference/rest/v1/projects.files/asyncBatchAnnotate)	**`POST /v1/{parent=projects/*}/files:asyncBatchAnnotate`** Run asynchronous image detection and annotation for a list of generic files, such as PDF files, which may contain multiple pages and multiple images per page.

REST Resource: v1.projects.images

(/vision/docs/reference/rest/v1/projects.images)

Methods	
`annotate` (/vision/docs/reference/rest/v1/projects.images/annotate)	`POST /v1/{parent=projects/*}/images:annotate` Run image detection and annotation for a batch of images.
`asyncBatchAnnotate` (/vision/docs/reference/rest/v1/projects.images/asyncBatchAnnotate)	`POST /v1/{parent=projects/*}/images:asyncBatchAnnotate` Run asynchronous image detection and annotation for a list of images.

REST Resource: v1.projects.locations.files

(/vision/docs/reference/rest/v1/projects.locations.files)

Methods	
`annotate` (/vision/docs/reference/rest/v1/projects.locations.files/annotate)	`POST /v1/{parent=projects/*/locations/*}/files:annotate` Service that performs image detection and annotation for a batch of files.
`asyncBatchAnnotate` (/vision/docs/reference/rest/v1/projects.locations.files/asyncBatchAnnotate)	`POST /v1/{parent=projects/*/locations/*}/files:asyncBatchAnnotate` Run asynchronous image detection and annotation for a list of generic files, such as PDF files, which may contain multiple pages and multiple images per page.

REST Resource: v1.projects.locations.images

(/vision/docs/reference/rest/v1/projects.locations.images)

Methods	
`annotate` (/vision/docs/reference/rest/v1/projects.locations.images/annotate)	`POST /v1/{parent=projects/*/locations/*}/images:annotate` Run image detection and annotation for a batch of images.

Methods	
asyncBatchAnnotate (/vision/docs/reference/rest/v1/projects.locations.images/asyncBatchAnnotate)	**`POST /v1/{parent=projects/*/locations/*}/images:asyncBatchAnnotate`** Run asynchronous image detection and annotation for a list of images.

REST Resource: v1.projects.locations.operations

(/vision/docs/reference/rest/v1/projects.locations.operations)

Methods	
get (/vision/docs/reference/rest/v1/projects.locations.operations/get)	**`GET /v1/{name=projects/*/locations/*/operations/*}`** Gets the latest state of a long-running operation.

REST Resource: v1.projects.locations.productSets

(/vision/docs/reference/rest/v1/projects.locations.productSets)

Methods	
addProduct (/vision/docs/reference/rest/v1/projects.locations.productSets/addProduct)	**`POST /v1/{name=projects/*/locations/*/productSets/*}:addProduct`** Adds a Product to the specified ProductSet.
create (/vision/docs/reference/rest/v1/projects.locations.productSets/create)	**`POST /v1/{parent=projects/*/locations/*}/productSets`** Creates and returns a new ProductSet resource.

Methods	
`delete` (/vision/docs/reference/rest/v1/projects.locations.productSets/delete)	`DELETE /v1/{name=projects/*/locations/*/productSets/*}` Permanently deletes a ProductSet.
`get` (/vision/docs/reference/rest/v1/projects.locations.productSets/get)	`GET /v1/{name=projects/*/locations/*/productSets/*}` Gets information associated with a ProductSet.
`import` (/vision/docs/reference/rest/v1/projects.locations.productSets/import)	`POST /v1/{parent=projects/*/locations/*}/productSets:import` Asynchronous API that imports a list of reference images to specified product sets based on a list of image information.
`list` (/vision/docs/reference/rest/v1/projects.locations.productSets/list)	`GET /v1/{parent=projects/*/locations/*}/productSets` Lists ProductSets in an unspecified order.
`patch` (/vision/docs/reference/rest/v1/projects.locations.productSets/patch)	`PATCH /v1/{productSet.name=projects/*/locations/*/productSets/*}` Makes changes to a ProductSet resource.
`removeProduct` (/vision/docs/reference/rest/v1/projects.locations.productSets/removeProduct)	`POST /v1/{name=projects/*/locations/*/productSets/*}:removeProduct` Removes a Product from the specified ProductSet.

REST Resource: v1.projects.locations.productSets.products

(/vision/docs/reference/rest/v1/projects.locations.productSets.products)

Methods	
list (/vision/docs/reference/rest/v1/projects.locations.productSets.products/list)	`GET /v1/{name=projects/*/locations/*/productSets/*}/products` Lists the Products in a ProductSet, in an unspecified order.

REST Resource: v1.projects.locations.products

(/vision/docs/reference/rest/v1/projects.locations.products)

Methods	
create (/vision/docs/reference/rest/v1/projects.locations.products/create)	`POST /v1/{parent=projects/*/locations/*}/products` Creates and returns a new product resource.
delete (/vision/docs/reference/rest/v1/projects.locations.products/delete)	`DELETE /v1/{name=projects/*/locations/*/products/*}` Permanently deletes a product and its reference images.
get (/vision/docs/reference/rest/v1/projects.locations.products/get)	`GET /v1/{name=projects/*/locations/*/products/*}` Gets information associated with a Product.
list (/vision/docs/reference/rest/v1/projects.locations.products/list)	`GET /v1/{parent=projects/*/locations/*}/products` Lists products in an unspecified order.
patch (/vision/docs/reference/rest/v1/projects.locations.products/patch)	`PATCH /v1/{product.name=projects/*/locations/*/products/*}` Makes changes to a Product resource.
purge (/vision/docs/reference/rest/v1/projects.locations.products/purge)	`POST /v1/{parent=projects/*/locations/*}/products:purge` Asynchronous API to delete all Products in a ProductSet or all Products that are in no ProductSet.

REST Resource: v1.projects.locations.products.referenceImages

(/vision/docs/reference/rest/v1/projects.locations.products.referenceImages)

Methods	
`create` (/vision/docs/reference/rest/v1/projects.locations.products.referenceImages/create)	`POST /v1/{parent=projects/*/locations/*/products/*}/referenceImages` Creates and returns a new ReferenceImage resource.
`delete` (/vision/docs/reference/rest/v1/projects.locations.products.referenceImages/delete)	`DELETE /v1/{name=projects/*/locations/*/products/*/referenceImages/*}` Permanently deletes a reference image.
`get` (/vision/docs/reference/rest/v1/projects.locations.products.referenceImages/get)	`GET /v1/{name=projects/*/locations/*/products/*/referenceImages/*}` Gets information associated with a ReferenceImage.
`list` (/vision/docs/reference/rest/v1/projects.locations.products.referenceImages/list)	`GET /v1/{parent=projects/*/locations/*/products/*}/referenceImages` Lists reference images.

REST Resource: v1.projects.operations

(/vision/docs/reference/rest/v1/projects.operations)

Methods

get (/vision/docs/reference/rest/v1/projects.operations/get)	`GET /v1/{name=projects/*/operations/*}` Gets the latest state of a long-running operation.

Last updated 2022-10-17 UTC.