Image to 3D API

Image to 3D API is a feature that allows you to integrate Meshy's Image to 3D capabilities into your own application. In this section, you'll find all the information you need to get started with this API.

`POST`   `/openapi/v1/image-to-3d`

Create an Image to 3D Task

This endpoint allows you to create a new Image to 3D task. Refer to The Image to 3D Task Object to see which properties are included with Image to 3D task object.

Parameters

Required attributes

`image_url`   `string`

Provide an image for Meshy to use in model creation. We currently support `.jpg`, `.jpeg`, and `.png` formats.

There are two ways to provide the image:

- **Publicly accessible URL**: A URL that is accessible from the public internet.

- **Data URI**: A base64-encoded data URI of the image. Example of a data URI: `data:image/jpeg;base64,<your base64-encoded image data>`.

Optional attributes

`ai_model`   `string`

ID of the model to use.

Available values:

- `meshy-4`

- `meshy-5`

Default to `latest` if not specified.

`topology`   string

Specify the topology of the generated model.

Available values:

- `quad` : Generate a quad-dominant mesh.
- `triangle` : Generate a decimated triangle mesh.

Default to `triangle` if not specified.

`target_polycount`   integer

Specify the target number of polygons in the generated model. The actual number of polygons may deviate from the target depending on the complexity of the geometry.

The valid value range varies depending on the user tier:

- 100 to 300,000 (inclusive)

Default to `30,000` if not specified.

`symmetry_mode`   string

The `symmetry_mode` field controls symmetry behavior during the model generation process.

The valid values are:

- `off` : Disables symmetry.
- `auto` : Automatically determines and applies symmetry based on input geometry.
- `on` : Enforces symmetry during generation.

Default to `auto` if not specified.

`should_remesh`   boolean

The `should_remesh` flag controls whether to enable the remesh phase.

When set to `false` , the API will directly return the highest-precision triangular mesh, ignoring `topology` and `target_polycount` .

Default to `true` if not specified.

`save_pre_remeshed_model` boolean

When set to `true`, Meshy also stores an extra GLB file before the remesh phase completes. Only takes effect when `should_remesh` is `true`.

Default to `false` if not specified.

`should_texture` boolean

The `should_texture` flag determines if textures are generated. Setting it to false skips the texture phase, providing a mesh without textures. This costs `20` credits for Meshy-6 models and `5` credits for other models. Setting it to `true` adds texture generation, costing an additional `10` credits. Default to `true` if not specified.

> ⓘ The parameters `enable_pbr`, `texture_prompt`, and `texture_image_url` all require `should_texture` to be `true` to function.

> ⓘ When `ai_model` is `latest` and `should_texture` is `true`, texturing uses Meshy-6-preview by default.

`enable_pbr` boolean

Generate PBR Maps (metallic, roughness, normal) in addition to the base color.

Default to `false` if not specified.

`pose_mode` string

Specify the pose mode for the generated model.

Available values:

- `a-pose` : Generate the model in an A pose.
- `t-pose` : Generate the model in a T pose.
- `""` (empty string): No specific pose applied.

Default to `""` (empty string) if not specified.

Deprecated. Use `pose_mode` instead.

Whether to generate the model in an A/T pose.

Default to `false` if not specified.

`texture_prompt`   string

Provide a text prompt to guide the texturing process. Maximum 600 characters.

`texture_image_url`   string

Provide a 2d image to guide the texturing process. We currently support `.jpg`, `.jpeg`, and `.png` formats.

There are two ways to provide the image:

- **Publicly accessible URL**: A URL that is accessible from the public internet

- **Data URI**: A base64-encoded data URI of the image. Example of a data URI: `data:image/jpeg;base64,<your base64-encoded image data>`

> ⓘ Image texturing may not work optimally if there are substantial geometry differences between the original asset and uploaded image. Only one of `texture_image_url` or `texture_prompt` may be used to guide the texturing process. If both parameters are provided, then `texture_prompt` will be used to texture the model by default. Texturing via either text or image will cost 10 credits per task.

`moderation`   boolean

When set to `true`, the input content will automatically be screened for potentially harmful content. If harmful content is detected, the task will not proceed to generation.

The content from the `image_url`, `texture_image_url`, and `texture_prompt` inputs will be screened.

Defaults to `false` if not specified.

Returns

Request

cURL **JavaScript** Python

POST /openapi/v1/image-to-3d

```javascript
import axios from 'axios'

const headers = { Authorization: `Bearer ${YOUR_API_KEY}` };
const payload = {
  // Using data URI example
  // image_url: 'data:image/png;base64,${YOUR_BASE64_ENCODED_IMAGE_DATA}',
  image_url: "<your publicly accessible image url or base64-encoded data URI>",
  enable_pbr: true,
  should_remesh: true,
  should_texture: true,
  save_pre_remeshed_model: true
};

try {
  const response = await axios.post(
    'https://api.meshy.ai/openapi/v1/image-to-3d',
    payload,
    { headers }
  );
  console.log(response.data);
} catch (error) {
  console.error(error);
}
```

Response

```json
{
  "result": "018a210d-8ba4-705c-b111-1f1776f7f578"
}
```

GET /openapi/v1/image-to-3d/:id

Retrieve an Image to 3D Task

This endpoint allows you to retrieve an Image to 3D task given a valid task `id`. Refer to **The Image to 3D Task Object** to see which properties are included with Image to 3D task object.

`id` path

Unique identifier for the Image to 3D task to retrieve.

Returns

The response contains the Image to 3D task object. Check **The Image to 3D Task Object** section for details.

Request   cURL **JavaScript** Python

```javascript
GET  /openapi/v1/image-to-3d/018a210d-8ba4-705c-b111-1f1776f7f578

import axios from 'axios'

const taskId = '018a210d-8ba4-705c-b111-1f1776f7f578';
const headers = { Authorization: `Bearer ${YOUR_API_KEY}` };

try {
  const response = await axios.get(
    `https://api.meshy.ai/openapi/v1/image-to-3d/${taskId}`,
    { headers }
  );
  console.log(response.data);
} catch (error) {
  console.error(error);
}
```

Response

```json
{
  "id": "018a210d-8ba4-705c-b111-1f1776f7f578",
  "type": "image-to-3d",
  "model_urls": {
    "glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "fbx": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "obj": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "usdz": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f
    "pre_remeshed_glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b1
  },
  "thumbnail_url": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f
  "texture_prompt": "",
  "progress": 100,
  "started_at": 1692771667037,
  "created_at": 1692771650657,
  "expires_at": 1692771679037,
  "finished_at": 1692771669037,
```

```
    {
      "base_color": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1
      "metallic": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1
      "normal": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f177
      "roughness": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f
    }
  ],
  "preceding_tasks": 0,
  "task_error": {
    "message": ""
  }
}
```

DELETE `/openapi/v1/image-to-3d/:id`

Delete an Image to 3D Task

This endpoint permanently deletes an Image to 3D task, including all associated models and data. This action is irreversible.

Path Parameters

`id` path

The ID of the Image to 3D task to delete.

Returns

Returns `200 OK` on success.

Request		cURL JavaScript Python

```
DELETE   /openapi/v1/image-to-3d/018a210d-8ba4-705c-b111-1f1776f7f578

import axios from 'axios'

const taskId = '018a210d-8ba4-705c-b111-1f1776f7f578'
const headers = { Authorization: `Bearer ${YOUR_API_KEY}` }

try {
  await axios.delete(
    `https://api.meshy.ai/openapi/v1/image-to-3d/${taskId}`,
    { headers }
```

```
  console.error(error)
}
```

Response

```
// Returns 200 Ok on success.
```

GET   `/v1/image-to-3d`

List Image to 3D Tasks

This endpoint allows you to retrieve a list of Image to 3D tasks.

Parameters

Optional attributes

`page_num`   integer

Page number for pagination. Starts and defaults to 1 .

`page_size`   integer

Page size limit. Defaults to 10 items. Maximum allowed is 50 items.

`sort_by`   string

Field to sort by. Available values:

- `+created_at` : Sort by creation time ascendly.
- `-created_at` : Sort by creation time descendly.

Returns

Returns a paginated list of The Image to 3D Task Objects.

Request	cURL	**JavaScript**	Python

GET   `/openapi/v1/image-to-3d`

```javascript
const headers = { Authorization: `Bearer ${YOUR_API_KEY}` };

try {
  const response = await axios.get(
    `https://api.meshy.ai/openapi/v1/image-to-3d?page_size=10`,
{ headers }
  );
  console.log(response.data);
} catch (error) {
  console.error(error);
}
```

Response

```json
[
  {
    "id": "018a210d-8ba4-705c-b111-1f1776f7f578",
    "type": "image-to-3d",
    "model_urls": {
      "glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7
      "fbx": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7
      "obj": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7
      "usdz": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f
      "pre_remeshed_glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-
    },
    "thumbnail_url": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-
    "texture_prompt": "",
    "progress": 100,
    "started_at": 1692771667037,
    "created_at": 1692771650657,
    "expires_at": 1692771679037,
    "finished_at": 1692771669037,
    "status": "SUCCEEDED",
    "texture_urls": [
      {
        "base_color": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111
        "metallic": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1
        "normal": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1
        "roughness": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-
      }
    ],
    "preceding_tasks": 0,
    "task_error": {
      "message": ""
    }
  }
]
```

`GET` `/openapi/v1/image-to-3d/:id/stream`

Stream an Image to 3D Task

This endpoint streams real-time updates for an Image to 3D task using Server-Sent Events (SSE).

Parameters

`id` path

Unique identifier for the Image to 3D task to stream.

Returns

Returns a stream of The Image to 3D Task Objects as Server-Sent Events.

For `PENDING` or `IN_PROGRESS` tasks, the response stream will only include necessary `progress` and `status` fields.

Request		cURL	JavaScript	Python

```
GET    /openapi/v1/image-to-3d/018a210d-8ba4-705c-b111-1f1776f7f578/stream

const eventSource = new EventSource(
  'https://api.meshy.ai/openapi/v1/image-to-3d/018a210d-8ba4-705c-b111-1f1776f7f
  {
    headers: { Authorization: `Bearer ${YOUR_API_KEY}` }
  }
);

eventSource.onmessage = (event) => {
  const data = JSON.parse(event.data);
  console.log(data);

  // Close stream when task is finished
  if (['SUCCEEDED', 'FAILED', 'CANCELED'].includes(data.status)) {
    eventSource.close();
  }
};

eventSource.onerror = (error) => {
  console.error('EventSource failed:', error);
  eventSource.close();
};
```

```
// Error event example
event: error
data: {
  "status_code": 404,
  "message": "Task not found"
}


// Message event examples illustrate task progress.
// For PENDING or IN_PROGRESS tasks, the response stream will not include all fi
event: message
data: {
  "id": "018a210d-8ba4-705c-b111-1f1776f7f578",
  "progress": 0,
  "status": "PENDING"
}


event: message
data: {
  "id": "018a210d-8ba4-705c-b111-1f1776f7f578",
  "type": "image-to-3d",
  "model_urls": {
  "glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f578
  "fbx": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f578
  "obj": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f578
  "usdz": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f57
  "pre_remeshed_glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111
  },
  "thumbnail_url": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f
  "texture_prompt": "",
  "progress": 100,
  "started_at": 1692771667037,
  "created_at": 1692771650657,
  "expires_at": 1692771679037,
  "finished_at": 1692771669037,
  "status": "SUCCEEDED",
  "texture_urls": [
    {
      "base_color": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1
      "metallic": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1
      "normal": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f177
      "roughness": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f
    }
  ],
  "preceding_tasks": 0,
  "task_error": {
    "message": ""
  }
}
```

The Image to 3D Task Object

The Image to 3D Task object is a work unit that Meshy keeps track of to generate a 3D model from an **image** input. The object has the following properties:

Properties

`id` string

Unique identifier for the task. While we use a k-sortable UUID for task ids as the implementation detail, you should **not** make any assumptions about the format of the id.

`type` string

Type of the Image to 3D task. The value is `image-to-3d`.

`model_urls` object

Downloadable URL to the textured 3D model file generated by Meshy. The property for a format will be omitted if the format is not generated instead of returning an empty string.

`glb` string

Downloadable URL to the GLB file.

`fbx` string

Downloadable URL to the FBX file.

`obj` string

Downloadable URL to the OBJ file.

`usdz` string

Downloadable URL to the USDZ file.

`mtl` string

Downloadable URL to the MTL file, returned alongside OBJ exports when textures are present.

`pre_remeshed_glb` string

ⓘ Available only when the task was created with both `should_remesh: true` and `save_pre_remeshed_model: true`.

ⓘ
- When `should_remesh` is `false`, `model_urls.glb` already corresponds to the original mesh and no `pre_remeshed_glb` is produced.
- When `should_remesh` is `true` and `save_pre_remeshed_model` is `true`, `model_urls.glb` stores the remeshed output while `model_urls.pre_remeshed_glb` stores the mesh captured before remeshing.

`thumbnail_url`   string

Downloadable URL to the thumbnail image of the model file.

`texture_prompt`   string

The text prompt that was used to guide the texturing process.

`texture_image_url`   string

Downloadable URL to the texture image that was used to guide the texturing process.

`progress`   integer

Progress of the task. If the task is not started yet, this property will be `0`. Once the task has succeeded, this will become `100`.

`started_at`   timestamp

Timestamp of when the task was started, in miliseconds. If the task is not started yet, this property will be `0`.

ⓘ A timestamp represents the number of milliseconds elapsed since January 1, 1970 UTC, following the RFC 3339 ↗ standard. For example, Friday, September 1, 2023 12:00:00 PM GMT is represented as `1693569600000`. This applies to **all** timestamps in Meshy API.

`created_at`   timestamp

`expires_at` timestamp

Timestamp of when the task result expires, in miliseconds.

`finished_at` timestamp

Timestamp of when the task was finished, in miliseconds. If the task is not finished yet, this property will be `0`.

`status` string

Status of the task. Possible values are one of `PENDING`, `IN_PROGRESS`, `SUCCEEDED`, `FAILED`, `CANCELED`.

`texture_urls` array

An array of texture URL objects that are generated from the task. Normally this only contains **one** texture URL object. Each texture URL has the following properties:

`base_color` string

Downloadable URL to the base color map image.

`metallic` string

Downloadable URL to the metallic map image.

> ⓘ If the task is created with `enable_pbr: false`, this property will be omitted.

`normal` string

Downloadable URL to the normal map image.

> ⓘ If the task is created with `enable_pbr: false`, this property will be omitted.

`roughness` string

Downloadable URL to the roughness map image.

> ⓘ If the task is created with `enable_pbr: false`, this property will be omitted.

The count of preceding tasks.

> ℹ The value of this field is meaningful only if the task status is `PENDING`.

`task_error` object

Error object that contains the error message if the task failed. The `message` property should be empty if the task succeeded.

`message` string

Detailed error message.

Example Image to 3D Task Object

```
{
  "id": "018a210d-8ba4-705c-b111-1f1776f7f578",
  "type": "image-to-3d",
  "model_urls": {
    "glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "fbx": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "obj": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "usdz": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f
    "pre_remeshed_glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b1
  },
  "thumbnail_url": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f
  "texture_prompt": "",
  "texture_image_url": "",
  "progress": 1,
  "started_at": 1692771667037,
  "created_at": 1692771650657,
  "expires_at": 1692771679037,
  "finished_at": 1692771669037,
  "status": "SUCCEEDED",
  "texture_urls": [
    {
      "base_color": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1
      "metallic": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1
      "normal": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f177
      "roughness": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f
    }
  ],
  "preceding_tasks": 0,
  "task_error": {
    "message": ""
  },
```

Multi-Image to 3D API

Multi-Image to 3D API is a feature that allows you to integrate Meshy's Multi-Image to 3D capabilities into your own application. In this section, you'll find all the information you need to get started with this API.

`POST`　　　`/openapi/v1/multi-image-to-3d`

Create a Multi-Image to 3D Task

This endpoint allows you to create a new Multi-Image to 3D task. Refer to The Multi-Image to 3D Task Object to see which properties are included with Multi-Image to 3D task object. Please note that Multi-Image to 3D's mesh generation is only available with the Meshy-5 model, while texture generation supports Meshy-6-preview model.

Parameters

Required attributes

`image_urls`　`array`

Provide 1 to 4 images for Meshy to use in model creation. We currently support `.jpg`, `.jpeg`, and `.png` formats. All images should depict the same object from different angles for best results.

There are two ways to provide each image:

- **Publicly accessible URL**: A URL that is accessible from the public internet.

- **Data URI**: A base64-encoded data URI of the image. Example of a data URI: `data:image/jpeg;base64,<your base64-encoded image data>`.

Optional attributes

`ai_model`　`string`

ID of the model to use.

- `meshy-5`
- `latest` : Meshy 6 Preview (texture)

Default to `latest` if not specified.

`topology` string

Specify the topology of the generated model.

Available values:

- `quad` : Generate a quad-dominant mesh.
- `triangle` : Generate a decimated triangle mesh.

Default to `triangle` if not specified.

`target_polycount` integer

Specify the target number of polygons in the generated model. The actual number of polygons may deviate from the target depending on the complexity of the geometry.

The valid value range varies depending on the user tier:

- 100 to 300,000 (inclusive)

Default to `30,000` if not specified.

`symmetry_mode` string

The `symmetry_mode` field controls symmetry behavior during the model generation process.

The valid values are:

- `off` : Disables symmetry.
- `auto` : Automatically determines and applies symmetry based on input geometry.
- `on` : Enforces symmetry during generation.

Default to `auto` if not specified.

`should_remesh` boolean

The `should_remesh` flag controls whether to enable the remesh phase.

Set to `true` if you want to toggle `topology` and `target_polycount`, which involves remeshing the initial model input.

Default to `true` if not specified.

`save_pre_remeshed_model` `boolean`

When set to `true`, Meshy also stores an extra GLB file before the remesh phase completes. Only takes effect when `should_remesh` is `true`.

Default to `false` if not specified.

`should_texture` `boolean`

The `should_texture` flag determines if textures are generated. Setting it to false skips the texture phase, providing a mesh without textures for `5` credits. Setting it to `true` adds texture generation, costing an additional `10` credits. Default to `true` if not specified.

> ⓘ The parameters `enable_pbr`, `texture_prompt`, and `texture_image_url` all require `should_texture` to be `true` to function.

> ⓘ When `should_texture` is `true`, texturing uses Meshy-6-preview by default while mesh generation continues to run on Meshy-5.

`enable_pbr` `boolean`

Generate PBR Maps (metallic, roughness, normal) in addition to the base color.

Default to `false` if not specified.

`pose_mode` `string`

Specify the pose mode for the generated model.

Available values:

- `a-pose` : Generate the model in an A pose.
- `t-pose` : Generate the model in a T pose.
- `""` (empty string): No specific pose applied.

`is_a_t_pose` boolean

Deprecated. Use `pose_mode` instead.

Whether to generate the model in an A/T pose.

Default to `false` if not specified.

`texture_prompt` string

Provide a text prompt to guide the texturing process. Maximum 600 characters.

`texture_image_url` string

Provide a 2d image to guide the texturing process. We currently support `.jpg`, `.jpeg`, and `.png` formats.

There are two ways to provide the image:

- **Publicly accessible URL**: A URL that is accessible from the public internet

- **Data URI**: A base64-encoded data URI of the image. Example of a data URI: `data:image/jpeg;base64,<your base64-encoded image data>`

> ℹ️ Image texturing may not work optimally if there are substantial geometry differences between the original asset and uploaded image. Only one of `texture_image_url` or `texture_prompt` may be used to guide the texturing process. If both parameters are provided, then `texture_prompt` will be used to texture the model by default. Texturing via either text or image will cost 10 credits per task.

`moderation` boolean

When set to `true`, the input content will automatically be screened for potentially harmful content. If harmful content is detected, the task will not proceed to generation.

Each image from `image_urls` and the text from `texture_prompt` will be screened.

Defaults to `false` if not specified.

Returns

Meshy

Open Web App

Request

cURL   **JavaScript**   Python

POST   /openapi/v1/multi-image-to-3d

```javascript
import axios from 'axios'

const headers = { Authorization: `Bearer ${YOUR_API_KEY}` };
const payload = {
  // Using data URI example
  // image_urls: [
  //    'data:image/png;base64,${YOUR_BASE64_ENCODED_IMAGE_DATA_1}',
  //    'data:image/png;base64,${YOUR_BASE64_ENCODED_IMAGE_DATA_2}'
  // ],
  image_urls: [
    "<your publicly accessible image url or base64-encoded data URI>",
    "<your second publicly accessible image url or base64-encoded data URI>"
  ],
  should_remesh: true,
  should_texture: true,
  save_pre_remeshed_model: true,
  enable_pbr: true
};

try {
  const response = await axios.post(
    'https://api.meshy.ai/openapi/v1/multi-image-to-3d',
    payload,
    { headers }
  );
  console.log(response.data);
} catch (error) {
  console.error(error);
}
```

Response

```json
{
  "result": "018a210d-8ba4-705c-b111-1f1776f7f578"
}
```

GET   /openapi/v1/multi-image-to-3d/:id

This endpoint allows you to retrieve a Multi-Image to 3D task given a valid task `id`. Refer to The Multi-Image to 3D Task Object to see which properties are included with Multi-Image to 3D task object.

Parameters

`id`   path

Unique identifier for the Multi-Image to 3D task to retrieve.

Returns

The response contains the Multi-Image to 3D task object. Check The Multi-Image to 3D Task Object section for details.

Request   cURL   **JavaScript**   Python

```javascript
GET  /openapi/v1/multi-image-to-3d/018a210d-8ba4-705c-b111-1f1776f7f578

import axios from 'axios'

const taskId = '018a210d-8ba4-705c-b111-1f1776f7f578';
const headers = { Authorization: `Bearer ${YOUR_API_KEY}` };

try {
  const response = await axios.get(
    `https://api.meshy.ai/openapi/v1/multi-image-to-3d/${taskId}`,
    { headers }
  );
  console.log(response.data);
} catch (error) {
  console.error(error);
}
```

Response

```json
{
  "id": "018a210d-8ba4-705c-b111-1f1776f7f578",
  "type": "multi-image-to-3d",
  "model_urls": {
    "glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "fbx": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "obj": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "usdz": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f
    "pre_remeshed_glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b1
```

```
  "texture_prompt": "",
  "progress": 100,
  "started_at": 1692771667037,
  "created_at": 1692771650657,
  "expires_at": 1692771679037,
  "finished_at": 1692771669037,
  "status": "SUCCEEDED",
  "texture_urls": [
    {
      "base_color": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1
      "metallic": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1
      "normal": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f177
      "roughness": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f
    }
  ],
  "preceding_tasks": 0,
  "task_error": {
    "message": ""
  }
}
```

DELETE   `/openapi/v1/multi-image-to-3d/:id`

Delete a Multi-Image to 3D Task

This endpoint permanently deletes a Multi-Image to 3D task, including all associated models and data. This action is irreversible.

Path Parameters

`id` path

The ID of the Multi-Image to 3D task to delete.

Returns

Returns `200 OK` on success.

Request	cURL	JavaScript	Python

DELETE   `/openapi/v1/multi-image-to-3d/018a210d-8ba4-705c-b111-1f1776f7f578`

```javascript
const taskId = '018a210d-8ba4-705c-b111-1f1776f7f578'
const headers = { Authorization: `Bearer ${YOUR_API_KEY}` }

try {
  await axios.delete(
    `https://api.meshy.ai/openapi/v1/multi-image-to-3d/${taskId}`,
    { headers }
  )
} catch (error) {
  console.error(error)
}
```

Response

```
// Returns 200 Ok on success.
```

`GET`   `/v1/multi-image-to-3d`

List Multi-Image to 3D Tasks

This endpoint allows you to retrieve a list of Multi-Image to 3D tasks.

Parameters

Optional attributes

`page_num`   integer

Page number for pagination. Starts and defaults to `1`.

`page_size`   integer

Page size limit. Defaults to `10` items. Maximum allowed is `50` items.

`sort_by`   string

Field to sort by. Available values:

- `+created_at` : Sort by creation time ascendly.

- `-created_at` : Sort by creation time descendly.

Returns a paginated list of The Multi-Image to 3D Task Objects.

	Request	cURL	JavaScript	Python

```
GET   /openapi/v1/multi-image-to-3d

import axios from 'axios'

const headers = { Authorization: `Bearer ${YOUR_API_KEY}` };

try {
 const response = await axios.get(
   `https://api.meshy.ai/openapi/v1/multi-image-to-3d?page_size=10`,
{ headers }
  );
 console.log(response.data);
} catch (error) {
 console.error(error);
}
```

Response

```
[
  {
    "id": "018a210d-8ba4-705c-b111-1f1776f7f578",
    "type": "multi-image-to-3d",
    "model_urls": {
      "glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7
      "fbx": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7
      "obj": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7
      "usdz": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f
      "pre_remeshed_glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-
    },
    "thumbnail_url": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-
    "texture_prompt": "",
    "progress": 100,
    "started_at": 1692771667037,
    "created_at": 1692771650657,
    "expires_at": 1692771679037,
    "finished_at": 1692771669037,
    "status": "SUCCEEDED",
    "texture_urls": [
      {
        "base_color": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111
        "metallic": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1
        "normal": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1
        "roughness": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-
      }
```

```
  "task_error": {
    "message": ""
  }
  }
]
```

GET `/openapi/v1/multi-image-to-3d/:id/stream`

Stream a Multi-Image to 3D Task

This endpoint streams real-time updates for a Multi-Image to 3D task using Server-Sent Events (SSE).

Parameters

`id` path

Unique identifier for the Multi-Image to 3D task to stream.

Returns

Returns a stream of The Multi-Image to 3D Task Objects as Server-Sent Events.

For `PENDING` or `IN_PROGRESS` tasks, the response stream will only include necessary `progress` and `status` fields.

Request	cURL	JavaScript	Python

```
GET   /openapi/v1/multi-image-to-3d/018a210d-8ba4-705c-b111-1f1776f7f578/stream

const eventSource = new EventSource(
  'https://api.meshy.ai/openapi/v1/multi-image-to-3d/018a210d-8ba4-705c-b111-1f1
  {
    headers: { Authorization: `Bearer ${YOUR_API_KEY}` }
  }
);

eventSource.onmessage = (event) => {
  const data = JSON.parse(event.data);
  console.log(data);

  // Close stream when task is finished
```

```
}
};


eventSource.onerror = (error) => {
console.error('EventSource failed:', error);
eventSource.close();
};
```

Response Stream

```
// Error event example
event: error
data: {
  "status_code": 404,
  "message": "Task not found"
}


// Message event examples illustrate task progress.
// For PENDING or IN_PROGRESS tasks, the response stream will not include all fi
event: message
data: {
  "id": "018a210d-8ba4-705c-b111-1f1776f7f578",
  "progress": 0,
  "status": "PENDING"
}


event: message
data: {
  "id": "018a210d-8ba4-705c-b111-1f1776f7f578",
  "type": "multi-image-to-3d",
  "model_urls": {
  "glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f578
  "fbx": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f578
  "obj": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f578
  "usdz": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f57
  "pre_remeshed_glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111
  },
  "thumbnail_url": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f
  "texture_prompt": "",
  "progress": 100,
  "started_at": 1692771667037,
  "created_at": 1692771650657,
  "expires_at": 1692771679037,
  "finished_at": 1692771669037,
  "status": "SUCCEEDED",
  "texture_urls": [
    {
      "base_color": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1
      "metallic": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1
```

```
      }
    ],
    "preceding_tasks": 0,
    "task_error": {
      "message": ""
    }
  }
}
```

The Multi-Image to 3D Task Object

The Multi-Image to 3D Task object is a work unit that Meshy keeps track of to generate a 3d model from multiple images (between 1 to 4 inclusive). The images should be of the same object, ideally from different views or angles. The object has the following properties:

Properties

`id` string

Unique identifier for the task. While we use a k-sortable UUID for task ids as the implementation detail, you should **not** make any assumptions about the format of the id.

`type` string

Type of the Multi-Image to 3D task. The value is `multi-image-to-3d`.

`model_urls` object

Downloadable URL to the textured 3D model file generated by Meshy. The property for a format will be omitted if the format is not generated instead of returning an empty string.

`glb` string

Downloadable URL to the GLB file.

`fbx` string

Downloadable URL to the FBX file.

`obj` string

`usdz` string

Downloadable URL to the USDZ file.

`mtl` string

Downloadable URL to the MTL file, returned alongside OBJ exports when textures are present.

`pre_remeshed_glb` string

Downloadable URL to the original GLB output before remeshing.

> ⓘ Available only when the task was created with both `should_remesh: true` and `save_pre_remeshed_model: true`.

> ⓘ
> - When `should_remesh` is `false`, `model_urls.glb` already corresponds to the original mesh and no `pre_remeshed_glb` is produced.
> - When `should_remesh` is `true` and `save_pre_remeshed_model` is `true`, `model_urls.glb` stores the remeshed output while `model_urls.pre_remeshed_glb` stores the mesh captured before remeshing.

`thumbnail_url` string

Downloadable URL to the thumbnail image of the model file.

`texture_prompt` string

The text prompt that was used to guide the texturing process.

`progress` integer

Progress of the task. If the task is not started yet, this property will be `0`. Once the task has succeeded, this will become `100`.

`started_at` timestamp

Timestamp of when the task was started, in miliseconds. If the task is not started yet, this property will be `0`.

> ⓘ A timestamp represents the number of milliseconds elapsed since January 1, 1970 UTC, following the RFC 3339 ↗ standard. For

Meshy API.

created_at `timestamp`

Timestamp of when the task was created, in miliseconds.

expires_at `timestamp`

Timestamp of when the task result expires, in miliseconds.

finished_at `timestamp`

Timestamp of when the task was finished, in miliseconds. If the task is not finished yet, this property will be `0`.

status `string`

Status of the task. Possible values are one of `PENDING`, `IN_PROGRESS`, `SUCCEEDED`, `FAILED`, `CANCELED`.

texture_urls `array`

An array of texture URL objects that are generated from the task. Normally this only contains **one** texture URL object. Each texture URL has the following properties:

base_color `string`

Downloadable URL to the base color map image.

metallic `string`

Downloadable URL to the metallic map image.

ⓘ If the task is created with `enable_pbr: false`, this property will be omitted.

normal `string`

Downloadable URL to the normal map image.

ⓘ If the task is created with `enable_pbr: false`, this property will be omitted.

roughness `string`

> ℹ If the task is created with `enable_pbr: false`, this property will be omitted.

`preceding_tasks` integer

The count of preceding tasks.

> ℹ The value of this field is meaningful only if the task status is `PENDING`.

`task_error` object

Error object that contains the error message if the task failed. The `message` property should be empty if the task succeeded.

`message` string

Detailed error message.

Example Multi-Image to 3D Task Object

```json
{
  "id": "018a210d-8ba4-705c-b111-1f1776f7f578",
  "type": "multi-image-to-3d",
  "model_urls": {
    "glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "fbx": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "obj": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f5
    "usdz": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1776f7f
    "pre_remeshed_glb": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b1
  },
  "thumbnail_url": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f
  "texture_prompt": "",
  "progress": 1,
  "started_at": 1692771667037,
  "created_at": 1692771650657,
  "expires_at": 1692771679037,
  "finished_at": 1692771669037,
  "status": "SUCCEEDED",
  "texture_urls": [
    {
      "base_color": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1
      "metallic": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f1
      "normal": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f177
      "roughness": "https://assets.meshy.ai/***/tasks/018a210d-8ba4-705c-b111-1f
```

    "task_error": {
      "message": ""
    },
}