

API文档

图像/视频超分、图像/视频臻彩、视频插帧、视频AI率检测与降低、图片/视频去人像 AI 味、去水印处理

通用说明

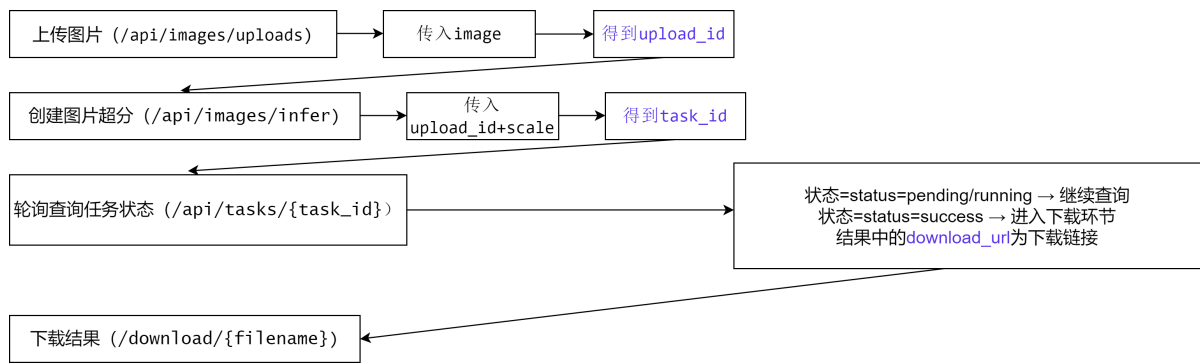
- Base URL 示例: `http://58.57.117.130:8006`
- 所有接口均使用 `multipart/form-data`
- 图片/视频超分、图片/视频臻彩、图片/视频去人像 AI 味、视频AI率检测与降低均使用公共上传接口: 图片为 `POST /api/images/uploads`, 表单字段为 `image`; 视频为 `POST /api/videos/uploads`, 表单字段为 `video`
- 视频插帧与去水印的上传接口分别包含插帧配置计算和自动遮罩检测等额外业务逻辑, 仍使用各自章节的专用上传接口
- 普通推理均为异步任务: 先上传获取 `upload_id`, 再创建任务获取 `task_id`, 最后轮询任务状态
- 返回中的 `download_url` 为相对路径, 使用时需拼接 Base URL
- 查询任务状态、下载结果、取消视频相关的任务使用的均是同一个API, 只是任务ID不同
- 视频上传时若首视频流不是 H.264 (如 HEVC/AV1, Chrome/Edge 无法预览), 服务端会自动额外转出一份 H.264 预览视频, 并在上传响应中返回可选字段 `preview_url` (仅页面预览用, 处理仍使用原文件); H.264 视频不返回该字段

目录

1. [图像超分](#)
2. [视频超分](#)
3. [图像臻彩](#)
4. [视频臻彩](#)
5. [视频插帧](#)
6. [视频AI率检测](#)
7. [视频AI率降低](#)
8. [去水印](#)
9. [图片去人像 AI 味](#)
10. [视频去人像 AI 味](#)

1. 图像超分

总体流程图



1.1 上传图片

POST `/api/images/uploads`

参数	类型	必填	说明
image	file	是	图片文件

支持: `.png`、`.jpg`、`.jpeg`、`.bmp`、`.webp`

```
curl -X POST "http://58.57.117.130:8006/api/images/uploads" -F "image=@input.png"
```

返回:

```
{
  "input_name": "input_720.jpg",
  "upload_id": "1788921795_537eb548"
}
```

1.2 创建图像超分任务

POST `/api/images/infer`

参数	类型	必填	说明
upload_id	string	是	图片上传返回的 <code>upload_id</code>
scale	string	否	<code>x2</code> , 默认 <code>x2</code>

```
curl -X POST "http://58.57.117.130:8006/api/images/infer" -F "upload_id=1788921795_537eb548" -F "scale=x2"
```

返回:

```
{
  "created_at": 1788921829,
  "download_url": "",
  "error": "",
  "input_name": "input_720.jpg",
  "media_type": "kmeans_sr_image",
  "output_name": "",
  "output_url": "",
}
```

```
"preview_url": "",
"progress": 0,
"scale": "x2",
"status": "pending",
"task_id": "1788921829_7d10a790",
"updated_at": 1788921829
}
```

1.3 查询任务状态

GET /api/tasks/{task_id}

```
curl "http://58.57.117.130:8006/api/tasks/你的task_id"
```

返回示例:

```
{
  "created_at": 1788921829,
  "download_url":
"/download/kmeans_sr_1788921829_7d10a790/input_720_x2_sr_1788921829.jpg",
  "elapsed_seconds": 0.55,
  "error": "",
  "inference_seconds": 0.38,
  "input_name": "input_720.jpg",
  "media_type": "kmeans_sr_image",
  "output_name": "input_720_x2_sr_1788921829.jpg",
  "output_url":
"/results/kmeans_sr_1788921829_7d10a790/input_720_x2_sr_1788921829.jpg",
  "preview_url":
"/results/kmeans_sr_1788921829_7d10a790/input_720_x2_sr_1788921829.jpg",
  "progress": 100,
  "scale": "x2",
  "status": "success",
  "task_id": "1788921829_7d10a790",
  "timing": {
    "file_end_to_end": 0.548359,
    "file_end_to_end_per_frame": 0.548359,
    "file_read": 0.006437,
    "file_write": 0.011968,
    "pure_model_forward": 0.374972,
    "pure_model_forward_per_frame": 0.374972,
    "single_inference_call": 0.382323,
    "single_inference_call_per_frame": 0.382323
  },
  "updated_at": 1788921829
}
```

status 取值: pending、running、success、failed、cancelled、waiting_mask。
success 代表可以下载结果了, cancelled 代表任务已经取消。

1.4 下载结果

GET /download/{filename}

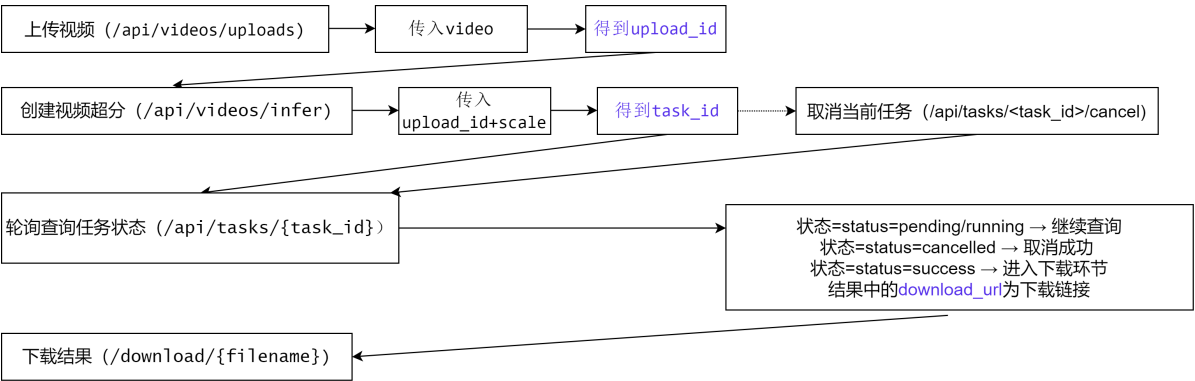
```
curl -O
"http://58.57.117.130:8006/download/kmeans_sr_1788921829_7d10a790/input_720_X2_SR_1788921829.jpg"
```

即：
http://58.57.117.130:8006/download/kmeans_sr_1788921829_7d10a790/input_720_X2_SR_1788921829.jpg

任务成功后，download_url 指向该接口，会以附件形式返回文件，参考返回示例中："download_url": "/download/filename"

2. 视频超分

总体流程图



2.1 上传视频

POST /api/videos/uploads

参数	类型	必填	说明
video	file	是	视频文件

支持：.mp4、.mov、.mkv、.avi、.webm

```
curl -X POST "http://58.57.117.130:8006/api/videos/uploads" -F "video=@input.mp4"
```

返回：

```
{
  "input_name": "720p.mp4",
  "metadata": {
    "duration": 5.07,
    "fps": 30.0,
    "frame_count": 152,
    "height": 720,
    "width": 1280
  },
  "upload_id": "1788921918_fa75fc33"
}
```

2.2 创建视频超分任务

POST /api/videos/infer

参数	类型	必填	说明
upload_id	string	是	视频上传返回的 <code>upload_id</code>
scale	string	否	<code>x2</code> , 默认 <code>x2</code>

```
curl -X POST "http://58.57.117.130:8006/api/videos/infer" -F
"upload_id=1788921918_fa75fc33" -F "scale=x2"
```

返回:

```
{
  "created_at": 1788921946,
  "download_url": "",
  "error": "",
  "input_name": "720p.mp4",
  "media_type": "kmeans_sr_video",
  "metadata": {
    "duration": 5.07,
    "fps": 30.0,
    "frame_count": 152,
    "height": 720,
    "width": 1280
  },
  "output_name": "",
  "output_url": "",
  "preview_url": "",
  "processed_frames": 0,
  "progress": 0,
  "scale": "x2",
  "status": "pending",
  "task_id": "1788921946_907449a1",
  "total_frames": 152,
  "updated_at": 1788921946
}
```

2.3 取消任务

POST /api/tasks/<task_id>/cancel

```
curl -X POST "http://58.57.117.130:8006/api/tasks/你的task_id/cancel"
```

返回示例:

```
{
  "status": "cancelling",
  "task_id": "1784738299_7b73c72b"
}
```

2.4 查询任务状态

GET /api/tasks/{task_id}

```
curl "http://58.57.117.130:8006/api/tasks/你的task_id"
```

返回示例:

```
{
  "created_at": 1788921946,
  "download_url": "/download/720p_x2_SR_1788921946.mp4",
  "elapsed_seconds": 14.33,
  "error": "",
  "inference_seconds": 12.81,
  "input_name": "720p.mp4",
  "media_type": "kmeans_sr_video",
  "metadata": {
    "duration": 5.07,
    "fps": 30.0,
    "frame_count": 152,
    "height": 720,
    "width": 1280
  },
  "output_name": "720p_x2_SR_1788921946.mp4",
  "output_url": "/results/720p_x2_SR_1788921946.mp4",
  "preview_url": "/results/720p_x2_SR_1788921946.mp4",
  "processed_frames": 152,
  "progress": 100,
  "scale": "x2",
  "status": "success",
  "task_id": "1788921946_907449a1",
  "timing": {
    "decoder": 0.334526,
    "decoder_per_frame": 0.002201,
    "encode": 2.49294,
    "encode_per_frame": 0.016401,
    "file_end_to_end_per_frame": 0.093307,
    "model_init": 7.5e-05,
    "pure_model_forward": 8.81806,
    "pure_model_forward_per_frame": 0.058014,
  }
}
```

```
    "single_inference_call": 12.810677,
    "single_inference_call_per_frame": 0.084281,
    "task_init": 0.546697,
    "total": 14.182616
  },
  "total_frames": 152,
  "updated_at": 1788921960
}
```

```
{
  "created_at": 1784819738,
  "download_url": "",
  "elapsed_seconds": 36.1,
  "error": "用户取消",
  "input_name": "onepiece_demo.mp4",
  "media_type": "video",
  "metadata": {
    "duration": 7.55,
    "fps": 23.98,
    "frame_count": 181,
    "height": 480,
    "width": 640
  },
  "output_name": "",
  "output_url": "",
  "outscale": "4",
  "preview_url": "",
  "processed_frames": 65,
  "progress": 35.9,
  "scale": "x4",
  "status": "cancelled",
  "task_id": "1784819738_9cc344aa",
  "total_frames": 181,
  "updated_at": 1784819776
}
```

`status` 取值: `pending`、`running`、`success`、`failed`、`cancelled`、`waiting_mask`。
`success` 代表可以下载结果了, `cancelled` 代表任务已经取消。

2.5 下载结果

GET `/download/{filename}`

```
curl -O http://58.57.117.130:8006/download/onepiece_demo_X2_SR_1784738299.mp4
```

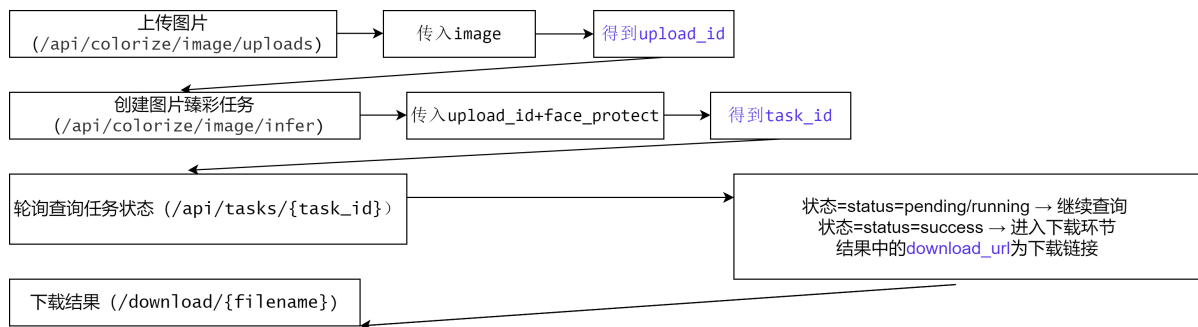
即: `http://58.57.117.130:8006/download/onepiece_demo_X2_SR_1784738299.mp4`

任务成功后, `download_url` 指向该接口, 会以附件形式返回文件, 参考返回示例

中: `"download_url": "/download/filename"`

3. 图像臻彩

总体流程图



3.1 上传图片

POST /api/images/uploads

参数	类型	必填	说明
image	file	是	图片文件

支持：.png、.jpg、.jpeg、.webp

```
curl -X POST "http://58.57.117.130:8006/api/images/uploads" -F "image=@input.png"
```

返回：

```
{
  "input_name": "1k.png",
  "upload_id": "1788922020_7572322f"
}
```

3.2 创建图像臻彩任务

POST /api/colorize/image/infer

参数	类型	必填	说明
upload_id	string	是	统一图片上传接口返回的 upload_id
face_protect	boolean/string	否	是否开启人脸肤色保护，默认 true

```
curl -X POST "http://58.57.117.130:8006/api/colorize/image/infer" -F
"upload_id=1788922020_7572322f" -F "face_protect=true"
```

返回：

```
{
  "created_at": 1788922045,
  "download_url": "",
  "error": "",
  "face_protect": true,
  "input_name": "1k.png",
  "media_type": "colorize_image",
  "output_name": "",
  "output_url": "",
}
```

```
"preview_url": "",
"progress": 0,
"status": "pending",
"task_id": "1788922045_282f399e",
"updated_at": 1788922045
}
```

3.3 查询任务状态

GET /api/tasks/{task_id}

```
curl "http://58.57.117.130:8006/api/tasks/你的task_id"
```

返回示例:

```
{
  "created_at": 1788922045,
  "download_url": "/download/colorize_img_1788922045_282f399e/1k.png",
  "elapsed_seconds": 0.36,
  "error": "",
  "face_protect": true,
  "inference_seconds": 0.36,
  "input_name": "1k.png",
  "media_type": "colorize_image",
  "output_name": "1k.png",
  "output_url": "/results/colorize_img_1788922045_282f399e/1k.png",
  "preview_url": "/results/colorize_img_1788922045_282f399e/1k.png",
  "progress": 100,
  "status": "success",
  "task_id": "1788922045_282f399e",
  "timing": {
    "file_end_to_end": 0.362284,
    "file_end_to_end_per_frame": 0.362284,
    "file_read": 0.018049,
    "file_write": 0.018049,
    "pure_model_forward_per_frame": 0.0,
    "pure_sdk_execution": 0.360978,
    "single_inference_call": 0.360978,
    "single_inference_call_per_frame": 0.360978
  },
  "updated_at": 1788922045
}
```

status 取值: pending、running、success、failed、cancelled、waiting_mask。
success 代表可以下载结果了, cancelled 代表任务已经取消。

3.4 下载结果

GET /download/{filename}

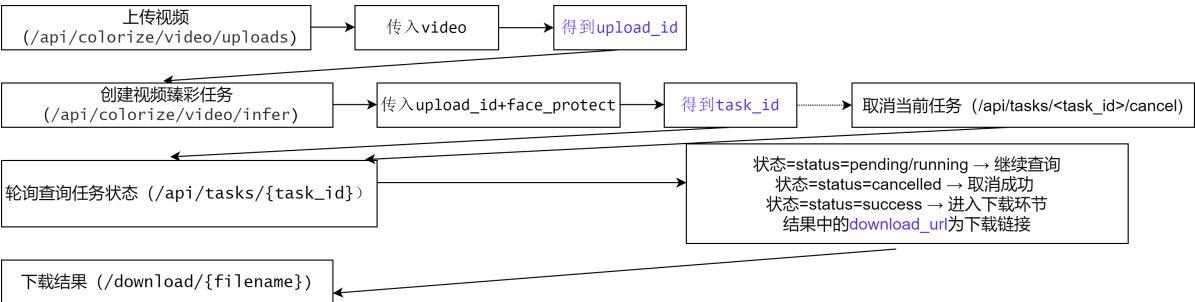
```
curl -o http://58.57.117.130:8006/download/000000000428_colorize_1784740146.jpg
```

即: http://58.57.117.130:8006/download/000000000428_colorize_1784740146.jpg

任务成功后，`download_url` 指向该接口，会以附件形式返回文件，参考返回示例中：`"download_url": "/download/filename"`

4. 视频臻彩

总体流程图



4.1 上传视频

POST `/api/videos/uploads`

参数	类型	必填	说明
video	file	是	视频文件

支持：`.mp4`、`.mov`、`.mkv`、`.avi`、`.webm`

```
curl -X POST "http://58.57.117.130:8006/api/videos/uploads" -F "video=@input.mp4"
```

返回：

```
{
  "input_name": "1-1k_5M.mp4",
  "metadata": {
    "duration": 31.79,
    "fps": 30.0,
    "frame_count": 953,
    "height": 1080,
    "width": 1920
  },
  "upload_id": "1788922091_9d30605d"
}
```

4.2 创建视频臻彩任务

POST `/api/colorize/video/infer`

参数	类型	必填	说明
upload_id	string	是	统一图片上传接口返回的 <code>upload_id</code>
face_protect	boolean/string	否	是否开启人脸肤色保护，默认 <code>true</code>

```
curl -X POST "http://58.57.117.130:8006/api/colorize/video/infer" -F
"upload_id=1784741016_aa87efbb" -F "face_protect=true"
```

返回:

```
{
  "created_at": 1788922127,
  "download_url": "",
  "error": "",
  "face_protect": true,
  "input_name": "1-1k_5M.mp4",
  "media_type": "colorize_video",
  "metadata": {
    "duration": 31.79,
    "fps": 30.0,
    "frame_count": 953,
    "height": 1080,
    "width": 1920
  },
  "output_name": "",
  "output_url": "",
  "preview_url": "",
  "processed_frames": 0,
  "progress": 0,
  "status": "pending",
  "task_id": "1788922127_9849bfcd",
  "total_frames": 953,
  "updated_at": 1788922127
}
```

4.3 取消任务

POST /api/tasks/<task_id>/cancel

```
curl -X POST "http://58.57.117.130:8006/api/tasks/你的task_id/cancel"
```

返回示例:

```
{
  "status": "cancelling",
  "task_id": "1784741147_717a8888"
}
```

4.4 查询任务状态

GET /api/tasks/{task_id}

```
curl "http://58.57.117.130:8006/api/tasks/你的task_id"
```

返回示例:

```
{
```

```

"created_at": 1788922127,
"download_url": "/download/colorize_vid_1788922127_9849bfcd/1-1k_5M.mp4",
"elapsed_seconds": 8.35,
"error": "",
"face_protect": true,
"inference_seconds": 8.35,
"input_name": "1-1k_5M.mp4",
"media_type": "colorize_video",
"metadata": {
  "duration": 31.79,
  "fps": 30.0,
  "frame_count": 953,
  "height": 1080,
  "width": 1920
},
"output_name": "1-1k_5M.mp4",
"output_url": "/results/colorize_vid_1788922127_9849bfcd/1-1k_5M.mp4",
"preview_url": "/results/colorize_vid_1788922127_9849bfcd/1-1k_5M.mp4",
"processed_frames": 0,
"progress": 100,
"status": "success",
"task_id": "1788922127_9849bfcd",
"timing": {
  "pure_sdk_execution": 8.345489645842463,
  "pure_sdk_execution_per_frame": 0.008757072031314232,
  "sdk_init": 0.00086195208132267,
  "single_inference_call": 8.345489645842463,
  "single_inference_call_per_frame": 0.008757072031314232,
  "total": 8.346882345154881,
  "total_per_frame": 0.008758533415692426
},
"total_frames": 953,
"updated_at": 1788922136
}

```

`status` 取值: `pending`、`running`、`success`、`failed`、`cancelled`、`waiting_mask`。
`success` 代表可以下载结果了, `cancelled` 代表任务已经取消。

4.5 下载结果

GET /download/{filename}

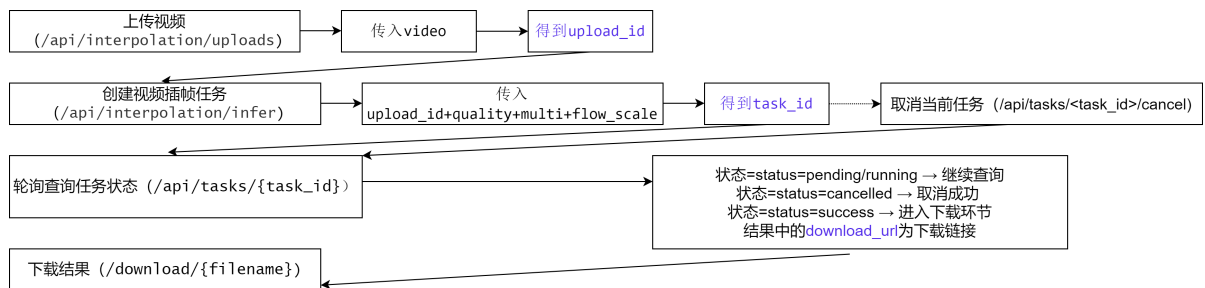
```
curl -o http://58.57.117.130:8006/download/480P_colorize_1784741147.mp4
```

即: `http://58.57.117.130:8006/download/480P_colorize_1784741147.mp4`

任务成功后, `download_url` 指向该接口, 会以附件形式返回文件, 参考返回示例
 中: `"download_url": "/download/filename"`

5. 视频插帧

总体流程图



5.1 上传插帧视频

POST `/api/interpolation/uploads`

参数	类型	必填	说明
video	file	是	待插帧视频

```
curl -X POST "http://58.57.117.130:8006/api/interpolation/uploads" -F
"video=@test.mp4"
```

返回：

```
{
  "input_name": "720p.mp4",
  "metadata": {
    "duration": 5.07,
    "fps": 30.0,
    "frame_count": 152,
    "height": 720,
    "width": 1280
  },
  "upload_id": "1788922178_858e1198"
}
```

5.2 创建视频插帧任务

POST `/api/interpolation/infer`

参数	类型	必填	说明
upload_id	string	是	插帧视频上传返回的 <code>upload_id</code>
output_fps	float	是	目标输出帧率，范围 1 ~ 240

```
curl -X POST "http://58.57.117.130:8006/api/interpolation/infer" \
-F "upload_id=1788316945_c98e0507" \
-F "output_fps=120"
```

返回：

```
{
  "created_at": 1788922210,
  "download_url": "",
}
```

```

"error": "",
"input_name": "720p.mp4",
"media_type": "interpolation",
"metadata": {
  "duration": 5.07,
  "fps": 30.0,
  "frame_count": 152,
  "height": 720,
  "width": 1280
},
"output_fps": 60.0,
"output_name": "",
"output_url": "",
"preview_url": "",
"processed_frames": 0,
"progress": 0,
"status": "pending",
"task_id": "1788922210_d1aafb73",
"total_frames": 152,
"updated_at": 1788922210
}

```

5.3 取消任务

POST /api/tasks/<task_id>/cancel

```
curl -X POST "http://58.57.117.130:8006/api/tasks/你的task_id/cancel"
```

返回示例:

```

{
  "status": "cancelling",
  "task_id": "1784738806_69b45a87"
}

```

5.4 查询任务状态

GET /api/tasks/{task_id}

```
curl "http://58.57.117.130:8006/api/tasks/你的task_id"
```

返回示例:

```

{
  "created_at": 1788922210,
  "download_url": "/download/720p_60FPS_INTERP_1788922210.mp4",
  "elapsed_seconds": 4.1,
  "error": "",
  "inference_seconds": 4.1,
  "input_name": "720p.mp4",
  "media_type": "interpolation",
  "metadata": {
    "duration": 5.07,

```

```

    "fps": 30.0,
    "frame_count": 152,
    "height": 720,
    "width": 1280
  },
  "output_fps": 60.0,
  "output_name": "720p_60FPS_INTERP_1788922210.mp4",
  "output_url": "/results/720p_60FPS_INTERP_1788922210.mp4",
  "preview_url": "/results/720p_60FPS_INTERP_1788922210.mp4",
  "processed_frames": 0,
  "progress": 100,
  "status": "success",
  "task_id": "1788922210_d1aafb73",
  "timing": {
    "pure_sdk_execution": 4.104786647949368,
    "sdk_init": 0.0004393099807202816,
    "single_inference_call": 4.104786647949368,
    "total": 4.104786647949368
  },
  "total_frames": 152,
  "updated_at": 1788922214
}

```

`status` 取值: `pending`、`running`、`success`、`failed`、`cancelled`、`waiting_mask`。
`success` 代表可以下载结果了, `cancelled` 代表任务已经取消。

5.5 下载结果

GET `/download/{filename}`

```
curl -O http://58.57.117.130:8006/download/720p_120FPS_INTERP_1788317175.mp4
```

即: `http://58.57.117.130:8006/download/720p_120FPS_INTERP_1788317175.mp4`

任务成功后, `download_url` 指向该接口, 会以附件形式返回文件, 参考返回示例
 中: `"download_url": "/download/filename"`

6. 视频AI率检测

6.1 上传视频

POST `/api/videos/uploads`

参数	类型	必填	说明
video	file	是	待检测视频

支持: `.mp4`、`.mov`、`.mkv`、`.avi`、`.webm`

```
curl -X POST "http://58.57.117.130:8006/api/videos/uploads" -F "video=@你的视频文件.mp4"
```

返回:

```
{
  "input_name": "13428404741794640.mp4",
  "metadata": {
    "duration": 5.04,
    "fps": 24.0,
    "frame_count": 121,
    "height": 1248,
    "width": 704
  },
  "preview_pending": true,
  "preview_url": "/uploads/1788922582_33aabebf/preview.mp4",
  "upload_id": "1788922582_33aabebf"
}
```

6.2 创建AI率检测任务

POST /api/deai-detect/infer

参数	类型	必填	说明
upload_id	string	是	POST /api/videos/uploads 返回的 upload_id

```
curl -X POST "http://58.57.117.130:8006/api/deai-detect/infer" -F
"upload_id=1784728000_abcd1234"
```

返回：

```
{
  "created_at": 1788922616,
  "detect_result": null,
  "download_url": "",
  "error": "",
  "input_name": "13428404741794640.mp4",
  "media_type": "videoadv_detect",
  "output_name": "",
  "output_url": "",
  "preview_url": "",
  "progress": 0,
  "status": "pending",
  "task_id": "1788922616_f6a72b04",
  "updated_at": 1788922616
}
```

6.3 查询任务状态

GET /api/tasks/{task_id}

```
curl "http://58.57.117.130:8006/api/tasks/你的task_id"
```

返回示例（成功时 status=success）：

```
{
```

```

"created_at": 1788922616,
"detect_result": {
  "continuous_score": 85.67922382688126,
  "decision_threshold": 0.4067833703866984,
  "probability_ai": 0.8840118971456496,
  "verdict": "AI"
},
"download_url": "",
"elapsed_seconds": 2.2,
"error": "",
"input_name": "13428404741794640.mp4",
"media_type": "videoadv_detect",
"output_name": "",
"output_url": "",
"preview_url": "",
"progress": 100,
"report_path":
"/data/shenzixuan/wt/web0908/web_results/videoadv_detect_1788922616_f6a72b04/detection_report.json",
"status": "success",
"task_id": "1788922616_f6a72b04",
"updated_at": 1788922618
}

```

`detect_result.continuous_score` 为该视频的 AI 率展示分（0~100，越大越可能是 AI 生成），是页面展示的主数值；`decision_threshold` 为判定阈值（约 0.4068），`probability_ai` 不低于阈值时判定为 AI 生成（以 `verdict` 为准）。

6.4 取消任务

POST `/api/tasks/{task_id}/cancel`

```
curl -X POST "http://58.57.117.130:8006/api/tasks/你的task_id/cancel"
```

返回示例：

```

{
  "status": "cancelling",
  "task_id": "1784739429_5e7f8f58"
}

```

`status` 取值：`pending`、`running`、`success`、`failed`、`cancelled`、`waiting_mask`。
`success` 代表检测完成，`cancelled` 代表任务已经取消。

7. 视频AI率降低

7.1 上传视频

POST `/api/videos/uploads`

参数	类型	必填	说明
video	file	是	待处理视频

支持：`.mp4`、`.mov`、`.mkv`、`.avi`、`.webm`

```
curl -X POST "http://58.57.117.130:8006/api/videos/uploads" -F "video=@你的视频文件.mp4"
```

返回：

```
{
  "input_name": "AIGC1.mp4",
  "metadata": {
    "duration": 5.04,
    "fps": 24.0,
    "frame_count": 121,
    "height": 832,
    "width": 1088
  },
  "upload_id": "1787133823_1c0438ce"
}
```

7.2 创建AI率降低任务

POST `/api/deai-attack/infer`

参数	类型	必填	说明
upload_id	string	是	<code>POST /api/videos/uploads</code> 返回的 <code>upload_id</code>

```
curl -X POST "http://58.57.117.130:8006/api/deai-attack/infer" \
-F "upload_id=1784728000_abcd1234"
```

返回：

```
{
  "attack_seed": 42001,
  "created_at": 1788922722,
  "detect_result": null,
  "download_url": "",
  "error": "",
  "input_name": "13428404741794640.mp4",
  "media_type": "videoadv_attack",
  "output_name": "",
  "output_url": "",
  "preview_url": "",
  "progress": 0,
  "status": "pending",
  "task_id": "1788922722_4ead1bb3",
  "updated_at": 1788922722
}
```

7.3 取消任务

POST /api/tasks/{task_id}/cancel

```
curl -X POST "http://58.57.117.130:8006/api/tasks/你的task_id/cancel"
```

返回示例:

```
{
  "status": "cancelling",
  "task_id": "1784739429_5e7f8f58"
}
```

7.4 查询任务状态

GET /api/tasks/{task_id}

```
curl "http://58.57.117.130:8006/api/tasks/你的task_id"
```

处理完成后 `status=success`, 返回 `elapsed_seconds` 与 `output_url / preview_url / download_url`:

```
{
  "attack_seed": 42001,
  "created_at": 1788922722,
  "detect_result": null,
  "download_url": "/download/13428404741794640_deai_1788922722.mp4",
  "elapsed_seconds": 8.5,
  "error": "",
  "input_name": "13428404741794640.mp4",
  "media_type": "videoadv_attack",
  "output_name": "13428404741794640_deai_1788922722.mp4",
  "output_url": "/results/13428404741794640_deai_1788922722.mp4",
  "preview_url": "/results/13428404741794640_deai_1788922722.mp4",
  "progress": 100,
  "status": "success",
  "task_id": "1788922722_4ead1bb3",
  "updated_at": 1788922731
}
```

7.5 下载结果

GET /download/{filename}

```
curl -O http://58.57.117.130:8006/download/test_ai_low_1786133327.mp4
```

任务成功后, `download_url` 指向该接口, 会以附件形式返回文件, 参考返回示例

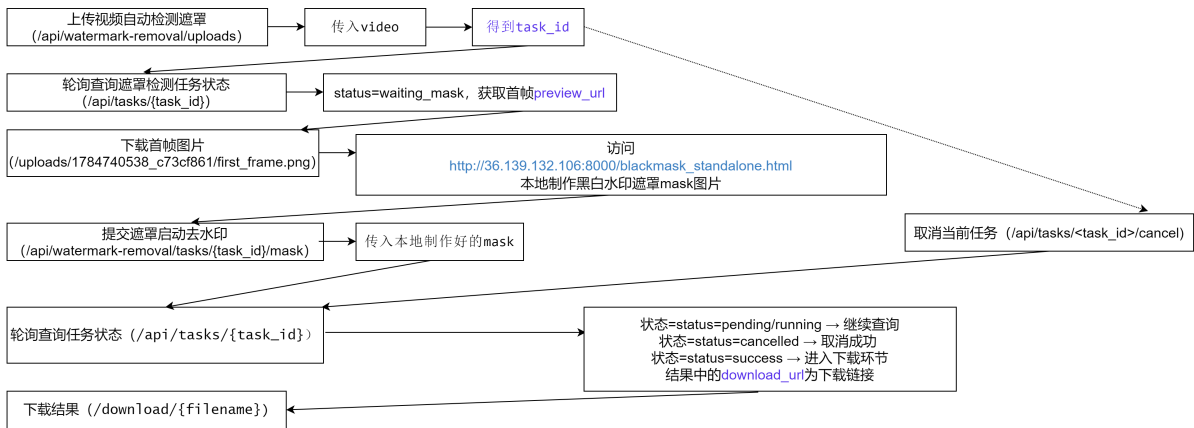
中: `"download_url": "/download/filename"`

`status` 取值: `pending`、`running`、`success`、`failed`、`cancelled`、`waiting_mask`。

`success` 代表可以下载结果了, `cancelled` 代表任务已经取消。

8. 去水印

总体流程图



8.1 上传视频并自动检测遮罩

POST `/api/watermark-removal/uploads`

参数	类型	必填	说明
video	file	是	待去水印视频

```
curl -X POST "http://58.57.117.130:8006/api/watermark-removal/uploads" -F "video=@input.mp4"
```

返回：

```
{
  "created_at": 1784824591,
  "download_url": "",
  "error": "",
  "input_name": "input.mp4",
  "media_type": "watermark_removal",
  "metadata": {
    "duration": 5.11,
    "fps": 24.0,
    "frame_count": 121,
    "height": 1248,
    "width": 704
  },
  "output_name": "",
  "output_url": "",
  "preview_url": "/uploads/1784824591_f2562ea1/first_frame.png",
  "video_preview_url": "/uploads/1784824591_f2562ea1/preview.mp4",
  "progress": 0,
  "status": "pending",
  "task_id": "1784824591_efd5110d",
  "updated_at": 1784824591
}
```

`video_preview_url` 为可选字段：仅当原视频首视频流不是 H.264 (HEVC/AV1 等) 时返回，指向服务端转出的 H.264 预览视频，供浏览器预览原视频；H.264 视频该字段为 `null`。
`preview_url` 仍为首帧图片（用于涂抹遮罩）。

8.2 查询去水印遮罩检测任务状态

GET `/api/tasks/{task_id}`

```
curl "http://58.57.117.130:8006/api/tasks/你的task_id"
```

返回示例：

```
{
  "auto_mask_path":
  "/data/szx/web3/web_uploads/1784824591_f2562ea1/auto_mask.png",
  "created_at": 1784824591,
  "download_url": "",
  "error": "",
  "input_name": "input.mp4",
  "media_type": "watermark_removal",
  "metadata": {
    "duration": 5.11,
    "fps": 24.0,
    "frame_count": 121,
    "height": 1248,
    "width": 704
  },
  "output_name": "",
  "output_url": "",
  "preview_url": "/uploads/1784824591_f2562ea1/first_frame.png",
  "progress": 10,
  "status": "waiting_mask",
  "task_id": "1784824591_efd5110d",
  "updated_at": 1784824592
}
```

8.3 提交遮罩并执行去水印任务

POST `/api/watermark-removal/tasks/{task_id}/mask`

参数	类型	必填	说明
mask	file	是	用户确认后的遮罩图片，推荐 PNG

遮罩图片是第一帧使用画笔遮住水印后的黑白mask。具体实现：

- 1) 7.1 上传视频并自动检测遮罩后返回的"preview_url":
"/uploads/1784740538_c73cf861/first_frame.png"为首帧图片，可使用(http://58.57.117.130:8006/uploads/1784807727_712c5a96/first_frame.png)下载到本地
- 2) 可访问(http://58.57.117.130:8006/blackmask_standalone.html)自行传入下载的first_frame图片，经过涂抹遮罩后可生成黑白mask并下载。

```
curl -X POST "http://58.57.117.130:8006/api/watermark-removal/tasks/1784728804_04de9693/mask" -F "mask=@first_frame_mask.png"
```

返回:

```
{
  "status": "pending",
  "task_id": "1784824591_efd5110d"
}
```

8.4 取消任务

POST /api/tasks/<task_id>/cancel

```
curl -X POST "http://58.57.117.130:8006/api/tasks/你的task_id/cancel"
```

返回示例:

```
{
  "status": "cancelling",
  "task_id": "1784824591_efd5110d"
}
```

8.5 查询去水印任务状态

GET /api/tasks/{task_id}

```
curl "http://58.57.117.130:8006/api/tasks/你的task_id"
```

返回示例:

```
{
  "auto_mask_path":
  "/data/szx/web3/web_uploads/1784824591_f2562ea1/auto_mask.png",
  "created_at": 1784824591,
  "download_url": "/download/input_watermark_removed_1784824591.mp4",
  "error": "",
  "input_name": "input.mp4",
  "media_type": "watermark_removal",
  "metadata": {
    "duration": 5.11,
    "fps": 24.0,
    "frame_count": 121,
    "height": 1248,
    "width": 704
  },
  "output_name": "input_watermark_removed_1784824591.mp4",
  "output_url": "/results/input_watermark_removed_1784824591.mp4",
  "preview_url": "/results/input_watermark_removed_1784824591.mp4",
  "progress": 100,
  "status": "success",
  "task_id": "1784824591_efd5110d",
}
```

```
"updated_at": 1784824705
}
```

`status` 取值: `pending`、`running`、`success`、`failed`、`cancelled`、`waiting_mask`。
`success` 代表可以下载结果了, `cancelled` 代表任务已经取消。

8.6 下载结果

GET /download/{filename}

```
curl -o http://58.57.117.130:8006/download/input_watermark_removed_1784824591.mp4
```

即: `http://58.57.117.130:8006/download/input_watermark_removed_1784824591.mp4`

任务成功后, `download_url` 指向该接口, 会以附件形式返回文件, 参考返回示例

中: `"download_url": "/download/filename"`

9. 图片去人像 AI 味

页面: **GET** /deai-person/image

图片上传使用统一接口 **POST** /api/images/uploads, 参数为 `image`。支持
.png、.jpg、.jpeg、.bmp、.webp; 上传响应仅返回 `upload_id` 与 `input_name`。

9.1 创建图像去AI味任务

POST /api/deai-person/image/infer

参数	类型	必填	说明
upload_id	string	是	<code>POST /api/images/uploads</code> 返回的 <code>upload_id</code>

```
curl -X POST "http://58.57.117.130:8006/api/deai-person/image/infer" -F
"upload_id=1784728000_abcd1234"
```

返回:

```
{
  "created_at": 1788922969,
  "download_url": "",
  "error": "",
  "input_name": "girl1_base_nolora.png",
  "media_type": "deai_person_image",
  "output_name": "",
  "output_url": "",
  "preview_url": "",
  "progress": 0,
  "status": "pending",
  "task_id": "1788922969_1590a2b8",
  "updated_at": 1788922969
}
```

默认最长边为 1024 像素、4 步推理，保留原图尺寸输出。

9.2 查询任务与下载结果

GET /api/tasks/{task_id}

```
curl "http://58.57.117.130:8006/api/tasks/你的task_id"
```

成功时 status 为 success，响应包含 output_url、preview_url、download_url 与 elapsed_seconds。结果图片可通过 GET /download/{filename} 下载。

返回示例：

```
{
  "created_at": 1788922969,
  "download_url": "/download/girl1_base_nolora_deai_1788922969.png",
  "elapsed_seconds": 5.34,
  "error": "",
  "input_name": "girl1_base_nolora.png",
  "media_type": "deai_person_image",
  "output_name": "girl1_base_nolora_deai_1788922969.png",
  "output_url": "/results/girl1_base_nolora_deai_1788922969.png",
  "preview_url": "/results/girl1_base_nolora_deai_1788922969.png",
  "progress": 100,
  "status": "success",
  "task_id": "1788922969_1590a2b8",
  "updated_at": 1788922975
}
```

10. 视频去人像 AI 味

页面：GET /deai-person/video

视频上传使用统一接口 POST /api/videos/uploads，参数为 video。支持 .mp4、.mov、.mkv、.avi、.webm；上传响应包含 upload_id、input_name 和视频 metadata，不包含 tile。

10.1 创建视频去AI味任务

POST /api/deai-person/video/infer

参数	类型	必填	说明
upload_id	string	是	POST /api/videos/uploads 返回的 upload_id

```
curl -X POST "http://58.57.117.130:8006/api/deai-person/video/infer" -F "upload_id=1784728000_abcd1234"
```

返回：

```
{
  "created_at": 1787134111,
```

```

"download_url": "",
"error": "",
"input_name": "AIGC1.mp4",
"media_type": "deai_person_video",
"metadata": {
  "duration": 5.04,
  "fps": 24.0,
  "frame_count": 121,
  "height": 832,
  "width": 1088
},
"output_name": "",
"output_url": "",
"preview_url": "",
"processed_frames": 0,
"progress": 0,
"status": "pending",
"task_id": "1787134111_ccb8d300",
"total_frames": 121,
"updated_at": 1787134111
}

```

10.2 取消任务

POST /api/tasks/{task_id}/cancel

```
curl -X POST "http://58.57.117.130:8006/api/tasks/1787123100_efgh5678/cancel"
```

返回：

```
{"task_id": "1787123100_efgh5678", "status": "cancelling"}
```

取消会在当前帧处理结束后生效，最终任务状态为 `cancelled`。

10.3 查询任务与下载结果

GET /api/tasks/{task_id}

```
curl "http://58.57.117.130:8006/api/tasks/你的task_id"
```

处理期间返回 `processed_frames`、`total_frames`、`progress` 和 `elapsed_seconds`。成功时 `status` 为 `success`，视频结果通过 `preview_url` 播放，通过 `download_url` 下载。

```

{
  "created_at": 1787134111,
  "download_url": "/download/AIGC1_deai_1787134111.mp4",
  "elapsed_seconds": 490.51,
  "error": "",
  "input_name": "AIGC1.mp4",
  "media_type": "deai_person_video",
  "metadata": {
    "duration": 5.04,
    "fps": 24.0,

```

```
    "frame_count": 121,  
    "height": 832,  
    "width": 1088  
  },  
  "output_name": "AIGC1_deai_1787134111.mp4",  
  "output_url": "/results/AIGC1_deai_1787134111.mp4",  
  "preview_url": "/results/AIGC1_deai_1787134111.mp4",  
  "processed_frames": 121,  
  "progress": 100,  
  "status": "success",  
  "task_id": "1787134111_ccb8d300",  
  "total_frames": 121,  
  "updated_at": 1787134604  
}
```