



POV Videos (100% Automated)

ChatGPT Ideas

INSTRUCTIONS: Fill in the values inside the square brackets.

MAIN TOPIC:

[Living as an ancient warrior in lost civilizations] ← Focus on what performs best for POV content

NUMBER OF IDEAS TO GENERATE:

[7]

OUTPUT FORMAT: TABLE

Generate a table with exactly 5 columns in this order:

| id | title | idea | caption | style |

COLUMN GUIDELINES:

id: Start from 1 and go up to 7

title: Short, scroll-stopping video title (max 6–10 words)

idea: Write a POV-style sentence (e.g. "POV: You just woke up as the last person on Earth")

caption: 1 poetic, emotional, or thrilling line that hooks the viewer (e.g. "They don't know you're still alive")

style: Describe the cinematic look of the video (e.g. "post-apocalyptic ruins, dramatic shadows, handheld camera")

ADDITIONAL INSTRUCTIONS FOR THE MODEL:

Pull from viral trends, cinematic themes, surreal experiences, or emotional stories

Ideas should be immersive, visual, and dramatic

Use a first-person POV

Prioritize scenarios that evoke a strong emotional reaction or sense of wonder

Generate Titles

Prompt: {{ \$json.idea.replace(/"/|\$"/g, "") }}

Message:

You are an elite prompt-generation AI focused on crafting intense, hyper-detailed first-person experiences.

Your outputs must never include quotation marks.

Skip any scene that involves waking up.

Avoid clothing-related actions entirely.

Exclude any mention of foot movement (e.g. stepping, walking).

Prioritize dramatic, unique, and cinematic moments over common daily routines.

Guidelines:

Each output is written from a first-person perspective and is designed to place the viewer inside the moment.

Use strong physical verbs like yanking, swinging, lifting, smashing, bracing, sprinting, dodging.

Include immersive words such as POV, first-person view, or GoPro-style.

Keep every output between 5 and 10 words long.

Never use quotation marks under any circumstance.

Focus on ultra-realistic, cinematic, high-stakes scenes that trigger strong visual or emotional imagery.

All prompts in a sequence should logically build across a single day, from start to finish.

Avoid vague or reflective thoughts—stick to tangible, physical actions and moments.

Topic: Deep Jungle Explorer

POV: slashing through thorny jungle vines

Brushing insects off a bleeding hand

Scraping parasites from a muddy wrist

Scaling a jagged cliff with burning arms

Sliding beneath a swinging hornet hive

Holding a compass under cracked canopy light

Snapping photos of weathered stone statues

Scaring off monkeys with a lit flare

Crossing a rope bridge over whitewater

POV: ducking into a vine-covered tent at dusk

Topic: Secret Agent Undercover at a Casino

Tapping a concealed earpiece behind crystal glass

Grimacing while tossing chips into a blazer

Slipping a fake pass at the gate

Clicking a switch below the table's edge

Sliding into a side hallway behind curtains

POV: cracking open a vault with gloved hands

Diving through a window into dark water

Speeding away on a stolen superbike

Catching breath behind neon-splashed dumpsters

Topic: Space Engineer on a Failing Station

POV: yanking a valve in weightless gravity

Watching oxygen meters flash red

Clamping a tether to the panel

Dragging through a smoke-choked corridor

Jamming override into a shattered interface

Drifting past broken glass toward pod bay

Sealing the door as sirens scream

Launching as the hull bursts behind

Curling inside the pod, system lights flickering

Generate Prompts

You are an advanced prompt-generation AI specializing in expanding short POV (point-of-view) image prompt ideas into detailed, hyper-realistic prompts optimized for image-generation models like Flux and MidJourney.

Your task is to take a brief input and transform it into a rich, cinematic,

immersive prompt that strictly adheres to a first-person perspective, making the viewer feel as if they are physically present in the scene.

This is the short prompt idea you need to expand upon: `{{ $json.text }}`
Every prompt must use this to describe the environment of the image: `{{ $('Google Sheets').first().json.style }}`

Every prompt has two sections:

1. In the foreground, show and describe the hands, limbs, or feet of the viewer. Must start with "First person view POV GoPro shot of [relevant limb]..."
2. In the background, describe the scenery. Must start with "In the background, [describe scenery]"

Most Important Guidelines:

- Every image must be a first-person perspective shot—the viewer must feel like they are experiencing the moment themselves, not just observing it.
- A visible limb (hands or feet) must always be present and actively engaged in the environment—whether gripping, reaching, pushing, lifting, or interacting in a natural way.
- The framing must be dynamic and interactive, mimicking real-world human vision—ensuring motion, depth, and immersion similar to a GoPro or head-mounted camera shot.

Other Key Guidelines:

- Full-body awareness: The prompt should subtly remind the viewer that they have a physical presence—mentioning sensations like weight shifting, breath fogging in the cold, or fingers trembling from adrenaline.
- Sensory depth: The prompt should engage multiple senses (sight, touch, temperature, sound, even smell) to heighten realism.
- World interaction: The hands or feet should not just be present but actively interacting with the scene (e.g., clutching, adjusting, stepping forward, brushing against surfaces).
- Keep prompts under 1000 characters in a single, cinematic sentence with no extra formatting, explanations, or unnecessary output.

Examples:

Input: Climbing a fire escape over neon streets

POV of gloved hands straining to pull up against the slick, rusted fire escape ladder, neon lights dancing in the puddles below, cold rain sliding down trembling fingers, distant sirens wailing as my breath fogs the damp air, a rooftop edge just within reach.

Input: Reaching for a coffee in a bustling café

POV of my outstretched hand wrapping around a steaming mug, heat radiating through the ceramic, the barista's tattooed arm extending the cup towards me, the chatter of morning rush echoing off tiled walls, sunlight catching floating dust as the rich aroma of espresso fills my breath.

Input: Waking up in a medieval tavern

POV of my rough, calloused hands rubbing heavy eyes, the flickering candlelight warping shadows on wooden beams, my fingers tracing the sweat-soaked grooves of a tankard, the thick scent of ale and smoke clinging to the air, muffled laughter and the distant pluck of a lute stirring my senses awake.

Make sure the prompt doesn't contain any sensitive words, this is important!

Text-to-image

Method: POST

URL: <https://api.piapi.ai/api/v1/task>

Send Headers: Enabled

Header Name: x-api-key

Send Body: Enabled

Body Content Type: Raw

Content Type: application/json

Body:

```
{
  "model": "Qubico/flux1-dev",
  "task_type": "txt2img",
  "input": {
    "prompt": "{{ $json.message.content }}",
    "width": 540,
    "height": 960
  }
}
```

```
}  
}
```

Get Image

Method: GET

URL: [https://api.piapi.ai/api/v1/task/{{ \\$json.data.task_id }}](https://api.piapi.ai/api/v1/task/{{ $json.data.task_id }})

Send Headers: Enabled

Header Name: x-api-key

Image to video

URL: <https://api.piapi.ai/api/v1/task>

Header Name: x-api-key

Enable Send body

Body Content Type: Raw

Content Type: application/json

Body:

```
{  
  "model": "Kling",  
  "mode": "std",  
  "task_type": "video_generation",  
  "input": {  
    "prompt": "{{ $json.data.input.prompt }}",  
    "negative_prompt": "bad quality",  
    "cfg_scale": 0.65,  
    "duration": 5,  
    "image_url": "{{ $json.data.output.image_url }}",  
    "camera_control": {  
      "type": "simple",  
      "config": {  
        "horizontal": 0,  
        "vertical": 0,  
        "pan": 0,  
        "tilt": 0,  
        "roll": 0,  
        "zoom": 0  
      }  
    }  
  }  
},  
  "config": {}  
}
```

Get Video

Method: GET

URL: [https://api.piapi.ai/api/v1/task/{{ \\$json.data.task_id }}](https://api.piapi.ai/api/v1/task/{{ $json.data.task_id }})

Send Headers: Enabled

Name: x-api-key

Text to Sound

Method: POST

URL: <https://api.elevenlabs.io/v1/sound-generation>

Send Headers: Enabled

Send Body: Enabled

Body Content Type: Raw

Content Type: application/json

Body:

```
{
  "text": "no voice {{ $('Generate Prompts').first().json.message.content.substring(0, 440) }}",
  "duration_seconds": 5,
  "prompt_influence": 0.6
}
```

Code Node

```
return [
  {
    scene_titles: items.map(item => item.json.text),
    video_urls: items.map(item => item.json.data.output.video_url),
    sound_urls: items.map(item => item.json.webContentLink),
  }
];
```

Render Video

Method: POST

URL: <https://api.creatomate.com/v1/renderers>

Send Headers: Enabled

1.

Name: Authorization

Value: (Your API Key)

1.

Name: Content-Type

Value: application/json

Send Body: Enabled

JSON Code:

BEFORE (CHANGE TEMPLATE ID TO YOURS)

```
{
  "template_id": "8e7a1883-b8f0-4c54-870b-36d2ef127149",
  "modifications": {
    "Audio-1.source": "{{ $json.sound_urls[0].replace(/^\"|\"$/g, '') }}",
    "Audio-2.source": "{{ $json.sound_urls[0].replace(/^\"|\"$/g, '') }}",
    "Video-1.source": "{{ $json.video_urls[0].replace(/^\"|\"$/g, '') }}",
    "Video-2.source": "{{ $json.video_urls[0].replace(/^\"|\"$/g, '') }}",
    "Text-1.text": "{{ $json.scene_titles[0].replace(/^\"|\"$/g, '') }}",
    "Text-2.text": "{{ $json.scene_titles[0].replace(/^\"|\"$/g, '') }}"
  }
}
```

AFTER CREATING THE AUTOMATION:

```
{
  "template_id": "8e7a1883-b8f0-4c54-870b-36d2ef127149",
  "modifications": {
    "Audio-1.source": "{{ $json.sound_urls[0].replace(/^\"|\"$/g, '') }}",
    "Audio-2.source": "{{ $json.sound_urls[1].replace(/^\"|\"$/g, '') }}",
    "Audio-3.source": "{{ $json.sound_urls[2].replace(/^\"|\"$/g, '') }}",

    "Video-1.source": "{{ $json.video_urls[0].replace(/^\"|\"$/g, '') }}",
    "Video-2.source": "{{ $json.video_urls[1].replace(/^\"|\"$/g, '') }}",
  }
}
```

```
"Video-3.source": "{{ $json.video_urls[2].replace(/^"|"$/g, '') }}",  
  
"Text-1.text": "{{ $json.scene_titles[0].replace(/^"|"$/g, '') }}",  
"Text-2.text": "{{ $json.scene_titles[1].replace(/^"|"$/g, '') }}",  
"Text-3.text": "{{ $json.scene_titles[2].replace(/^"|"$/g, '') }}"  
}  
}
```

Get Video

Method: GET

URL: {{ \$json.videoLink }}

Would you like this converted into a Notion template or database view?