AI Video Generation

AI Video Generation Timeline

- **Early 2023 –Low resolution videos, melted textures, 3 second limits**

AI Video Generation Timeline

- 2024 – OpenAI releases Sora – 60 second photorealistic clips with complex physics are possible
  - Remained in closed beta
- Google Veo, Kling 1.0, and Luma Dream Machine launch
  - Publicly accessible models

AI Video Generation Timeline

- **2025 – Diffusion Transformers (DiT) start to understand gravity, fluid dynamics, and “Character Lock”**
- **Google Veo 3, Sora 2, Runway Gen-3 Alpha, Haiper 2.0, Kling 2.6, Luma Ray 2**

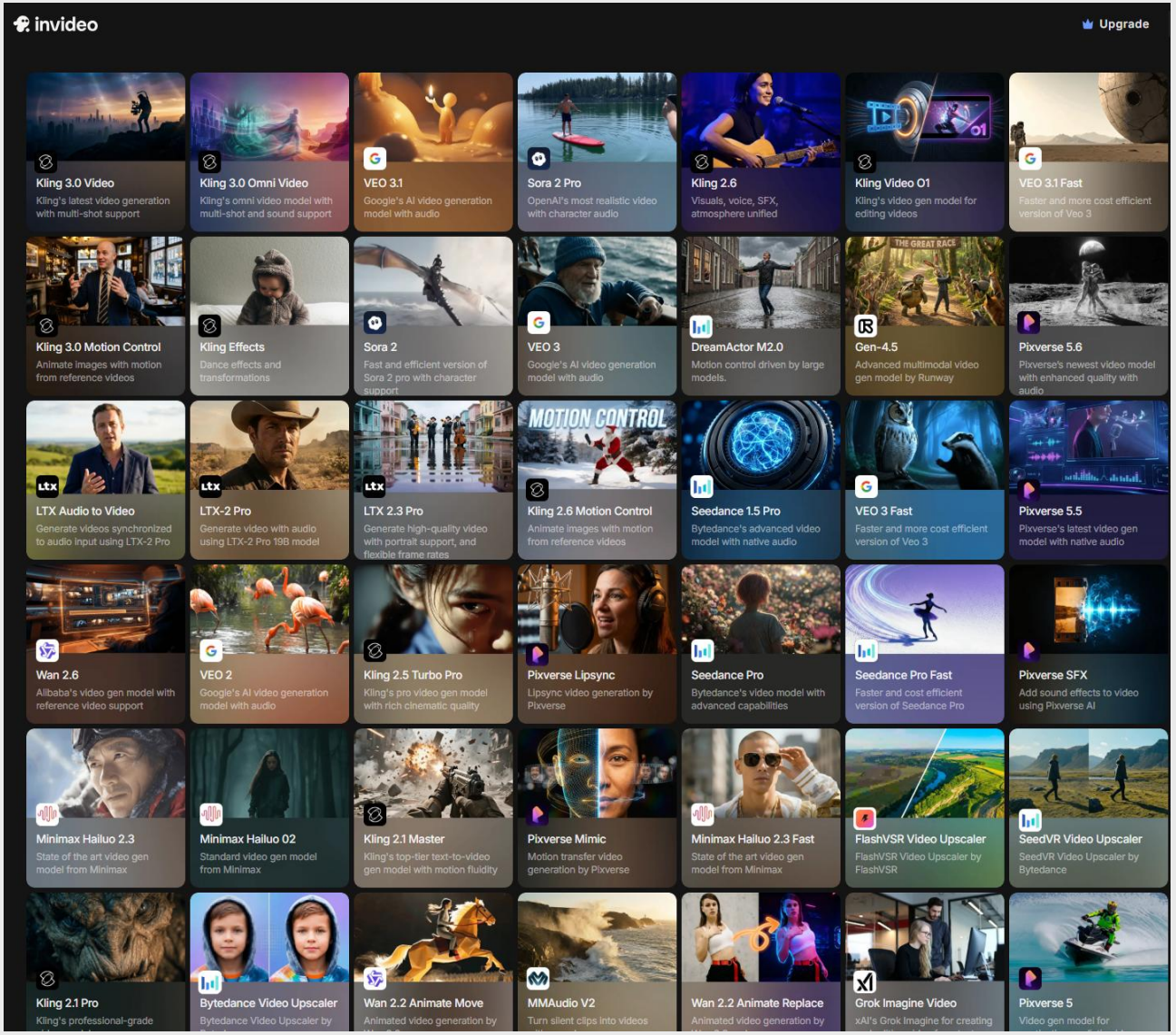
2025: Sora 2 vs Google Veo 3

AI Video Generation Timeline

- **2026 – AI video models generate hyper realistic longer duration videos with audio and consistent characters**

- **Google Veo 3.1, OpenAI Sora 2, ByteDance Seedance 2.0, xAI Grok Imagine, Alibaba Wan 2.6**

AI Video Models 2026



AI Video Models 2026

Model	Max Duration	Max Resolution	Native Audio	API Cost (Per Min)	Best For
Veo 3.1	60s+ (Chained)	**4K Ultra HD**	Yes (Sync'd)	**$45.00**	Cinematic quality & native sound
Runway Gen-4.5	16s (Loopable)	4K	No (Lip-sync)	**$24.00**	Professional control (Motion Brush)
Sora 2 (Pro)	25s	1080p	Yes (Sync'd)	**$18.00**	Physical realism & Character Likeness
Kling 3.0	10s	**4K Ultra HD**	Yes	**$10.20**	High-speed action & E-commerce
Seedance 2.0	15s	2K	Yes	**$9.00**	Multi-shot sequences & CapCut users
Grok Imagine 1.0	15s	720p	Yes (Dialog)	**$3.00**	Viral social clips & "Spicy" mode
Wan 2.6	15s	1080p	No	**Free** (Local)	Open-source & Private data

Will Smith Eating
Spaghetti Progression

2023 2024 2025 2026

"WILL SMITH EATING SPAGHETTI"
BY SEEDANCE 2.0

AI and Hollywood

- AI video generation represents a potential disruption for film studios
- Some studios are embracing AI

InterPositive

- Not a “text-to-video” product
- Build proprietary AI models trained on your film footage
  - Digital cleanup of shots
  - Shot recovery
  - Relighting and reframing shots
- Remove the “drudgery” of filmmaking
- Ben Affleck is helping integrate these AI tools across Netflix’s global production pipeline
  - Netflix can reduce production time -> new episodes faster
  - Netflix can reduce production cost -> lower subscription fee?

AI Screenwriting

- **Ben Affleck told Joe Rogan “If you try to get ChatGPT or Claude or Gemini to write you something, it's really shtty. And it’s shtty because by its nature it goes to the mean, to the average. It’s not reliable.”**

- **However, there are AI models being tested that can write scripts and even generate movie ideas for studios**


POWERING THE FUTURE OF STORYTELLING.
AWESOME-O
AI. GENERATIVE CINEMA. REIMAGINED.
WWW.AWESOME-O.AI
© 2026 Awesome-O Entertainment Inc.

Video Generation APIs

- Gemini API key – Veo 3.1 videos
- Runway API key – Gen-4.5 and Veo videos
- Kling API key – Kling Omni 3.0 videos

Image Inputs for Veo

- **Veo models can take different types of image inputs**
  - **image – the first image in the scene**
  - **last_frame – the last image in the scene**
  - **reference_images - a list of up to 3 images (Veo 3.1 only) to preserve in the video (person, character, or product)**
- **Veo 3.1 can take video inputs to extend scenes**
  - **video – the video of the previous scene to continue**

Veo API

Input parameters (instances)

Parameter	Type	Description
`prompt`	string	Text description. Supports audio cues (dialogue in quotes, SFX, ambience).
`image`	Image	First frame to animate (image-to-video).
`last_frame`	Image	Ending frame for interpolation. Use with `image`.
`reference_images`	list	Up to 3 images to guide content (person/product). Veo 3.1 only.
`video`	Video	Prior Veo-generated video for extension. Veo 3.1 only.

Veo API

model

Value	Description
`veo-3.1-generate-preview`	Full Veo 3.1, best quality
`veo-3.1-fast-generate-preview`	Faster, optimized for speed
`veo-2.0-generate-001`	Veo 2 (stable)

config.aspect_ratio

Value	Description
`"16:9"`	Landscape (default)
`"9:16"`	Portrait (vertical)

config.duration_seconds

Value	Notes
`4`	Shortest
`6`	—
`8`	Required when using extension, reference images, 1080p, or 4k

config.resolution

Value	Notes
`"720p"`	Default. Required for video extension.
`"1080p"`	8s duration only. 16:9 only for Veo 3.
`"4k"`	8s duration only. Higher latency and cost.

Generating People with Veo

config.person_generation

> Region-dependent. In EU, UK, CH, MENA: Veo 3 allows `allow_adult` only.

Value	Description
`allow_all`	Allow all person generation
`allow_adult`	Adults only (required for image-to-video in some regions)
`dont_allow`	No people (Veo 2 only)

Veo API Code

```python
import time
from google import genai
from google.genai import types

client = genai.Client(api_key="YOUR_GEMINI_API_KEY")

# First frame for image-to-video (optional)
first_frame = types.Image.from_file(location="path/to/image.png")

# Reference images (up to 3) — Veo 3.1 only, preserves subject appearance
ref1 = types.VideoGenerationReferenceImage(
    image=types.Image.from_file(location="path/to/person.png"),
    reference_type="asset",
)

operation = client.models.generate_videos(
    model="veo-3.1-generate-preview",
    prompt="A woman walks through a sunlit garden, gentle breeze.",
    image=first_frame,  # Optional: animate this as first frame
    config=types.GenerateVideosConfig(
        aspect_ratio="9:16",       # "16:9" or "9:16"
        duration_seconds=8,        # 4, 6, or 8
        resolution="720p",         # "720p", "1080p", "4k"
        number_of_videos=1,
        reference_images=[ref1],   # Optional: up to 3 (Veo 3.1 only)
        # person_generation="allow_adult",  # Optional: region-dependent
    ),
)
```

Veo API Code for Video Extension

```
# Extends a previously generated Veo video by ~7 seconds.
# Input must be from a prior generate_videos() call.
# Extension limited to 720p.

operation = client.models.generate_videos(
    model="veo-3.1-generate-preview",
    prompt="Track the scene as it continues into the garden.",
    video=previous_operation.response.generated_videos[0].video,
    config=types.GenerateVideosConfig(
        number_of_videos=1,
        resolution="720p",  # Required: 720p only for extension
    ),
)
```

Veo API Code for Interpolation (First + Last Frame)

```
# Generate video between two images.
operation = client.models.generate_videos(
    model="veo-3.1-generate-preview",
    prompt="Smooth transition from the first to the last frame.",
    image=first_frame_image,
    config=types.GenerateVideosConfig(
        last_frame=last_frame_image,
    ),
)
```

Runway API

- **Runway lets you use Runway and Veo models (and make 50 videos per day)**

model (Image-to-Video)

Value	Description
`gen4_turbo`	Fast, cost-effective (5 credits/sec)
`gen4.5`	Higher quality (12 credits/sec)
`gen3a_turbo`	Older model
`veo3.1`	Google Veo via Runway
`veo3.1_fast`	Faster Veo
`veo3`	Veo 3

Runway API

ratio (Image-to-Video)

Value	Description
`1280:720`	Landscape 16:9
`720:1280`	Portrait 9:16 (vertical)
`1104:832`	4:3
`960:960`	Square 1:1
`832:1104`	3:4 portrait
`1584:672`	Ultrawide

duration

Value	Description
2–10	Integer seconds. Longer = more credits.

prompt_image (Image-to-Video)

Format	Description
`HTTPS URL`	Public image URL
`runway://...`	Runway upload URI (from uploads API)
`data:image/...`	Base64 data URI
Array	`[{"uri": "...", "position": "first"}]` for positioned frames

Content Moderation in Runway API

- We need to set the public_figure_threshold to low if we want to make fun celebrity videos

content_moderation	object	{"public_figure_threshold": "auto"|"low"}

Runway API Cost

Credit costs (approx.)

Model	Credits/sec	5 sec video
gen4_turbo	5	~25 credits
gen4.5	12	~60 credits
act_two	5	~25 credits
veo3.1_fast	15	~75 credits
veo3 / veo3.1	40	~200 credits

Runway API Code

```python
from runwayml import RunwayML

client = RunwayML(api_key="YOUR_RUNWAY_API_KEY")

# Image: HTTPS URL, Runway URI (runway://...), or data URI (..."  # or URL

task = client.image_to_video.create(
    model="gen4_turbo",
    prompt_image=image_uri,
    prompt_text="Camera slowly pans across the scene, soft natural light.",
    ratio="720:1280",        # Output resolution
    duration=8,              # 2-10 seconds
    seed=42,                 # Optional: for reproducibility
    # content_moderation={"public_figure_threshold": "low"},  # Optional
)

result = task.wait_for_task_output(timeout=600)
video_url = result.output[0]
```

Kling API

- **Kling Omni model has many of the same features as Veo 3.1**

Inputs	Workflow
`frame_start` (+ optional `frame_end`)	**Frames** — first and last frame interpolation
`video_1` + `video_mode=transform`	**Video Transform** — modify a reference video
`video_1` + `video_mode=reference` (default)	**Video Reference** — style guided by video
Images/elements only, or text-only	**Default** — text-to-video with optional refs

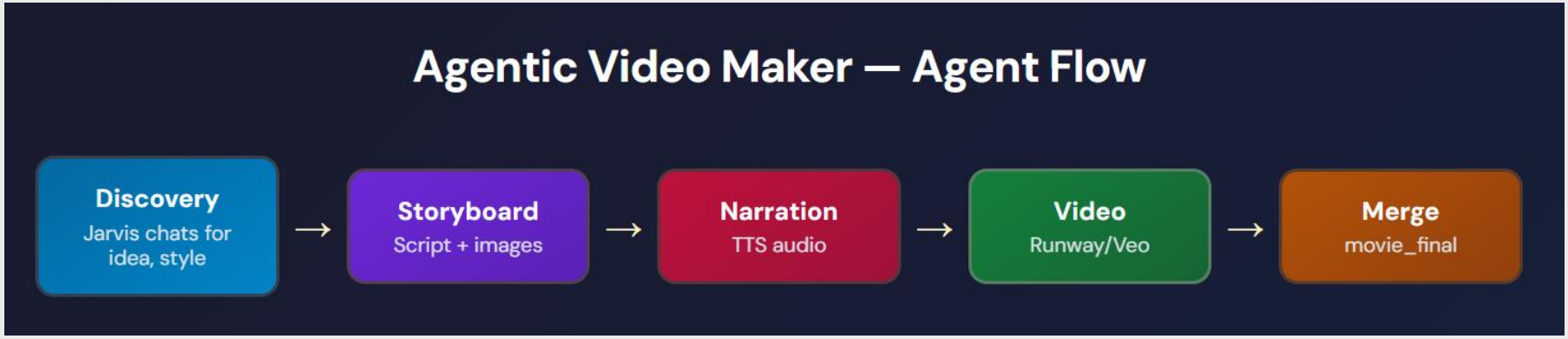
Kling Omni Multi-Shot

- **Kling's Omni v3 model can do multiple shots for a single scene**

```
# Split into multiple shots with separate prompts
# Min 2 shots, total duration 3-15 seconds
response = requests.post(API_URL, headers=headers, json={
    "omni_version": "v3",
    "shot_1_prompt": "A cat walks into a room",
    "shot_1_duration": "3",
    "shot_2_prompt": "The cat jumps onto the couch",
    "shot_2_duration": "4",
})
# Cannot use prompt or duration with multi-shot
```

Agentic AI Video Generation

- We can create agents that produce videos for us
- The audio is a narration only
- Generate images to anchor each scene video



Coding Session

- Make a Pydantic agent to create AI videos
- Storyboard agent – chat with agent to create images, descriptions, and narrations for a video storyboard
  - Allow the use of multiple anchor images to put characters or scenes in video
- Production agents – make audio narration and movie files for scenes, merge files to make final video