



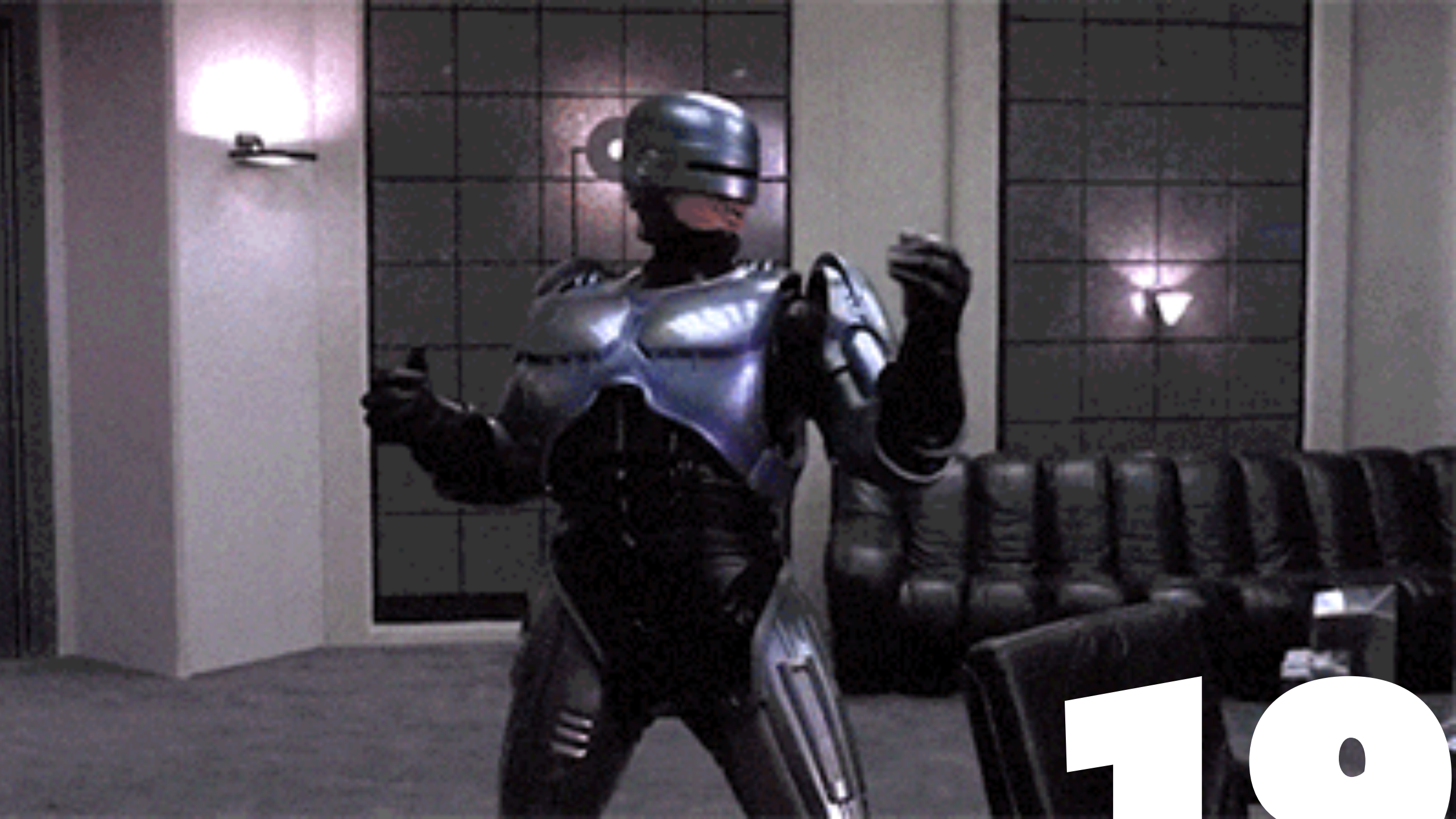
TYLER
STICKA!

GIFs are forever...

Let's make them better!

CascadiaJS 2024





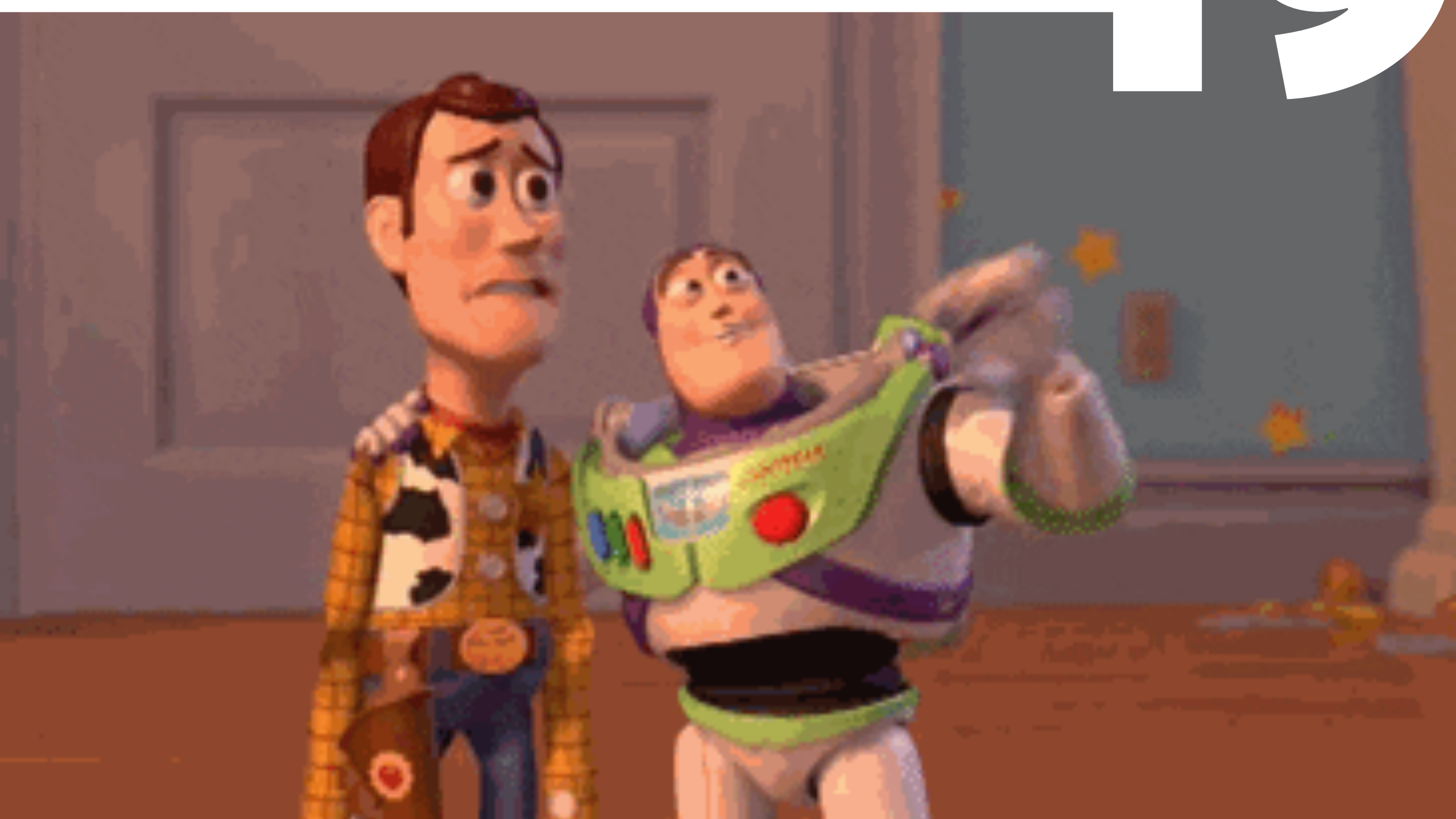
1987.





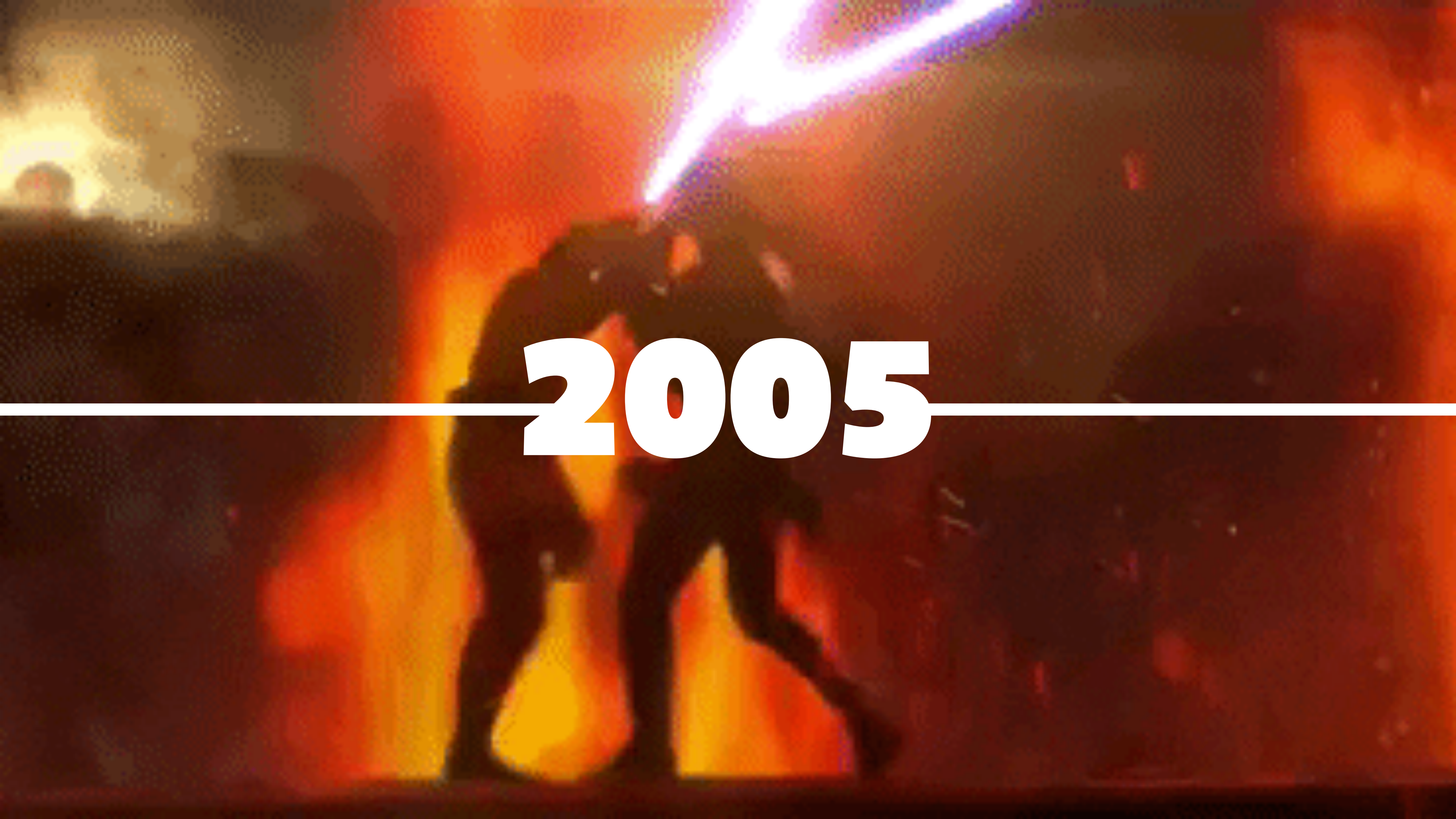
1995

UGH, AS IF!



NETSCAPE
Now! 2.0

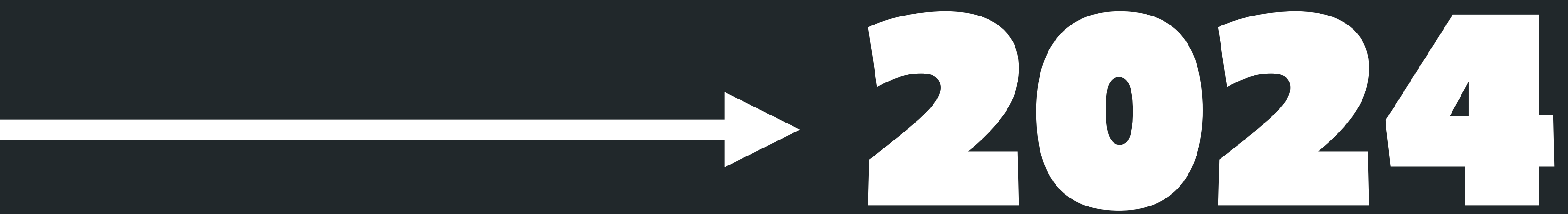
What YEAR is it?

A person in a dark suit is captured in a dynamic, dancing pose. The background is a vibrant, abstract scene with warm, glowing colors like orange, red, and yellow, suggesting a stage or a club environment. A bright, purple and white light source is visible at the top, creating a dramatic effect. The overall atmosphere is energetic and artistic.

2005



2015

A white arrow points from the left edge of the frame towards the year '2024'. The arrow is a simple, solid white line with a triangular arrowhead. The year '2024' is written in a large, bold, white sans-serif font, positioned to the right of the arrow's tip. The entire graphic is centered vertically on a dark gray background.

2024



2024









256 Colors Per Frame?



~~256 Colors Per Frame?~~



~~256 Colors Per Frame?~~

Single-Bit Transparency?



~~256 Colors Per Frame?~~

~~Single-Bit Transparency?~~



~~256 Colors Per Frame?~~

~~Single-Bit Transparency?~~

Image Sequences?



256 Co

Single



Image Sequences?





“Using emoji, emoticons and GIFs in a texted conversation instantly signals the difference between sincerity and a joke or sarcasm.”

— **Jenna Wortham**

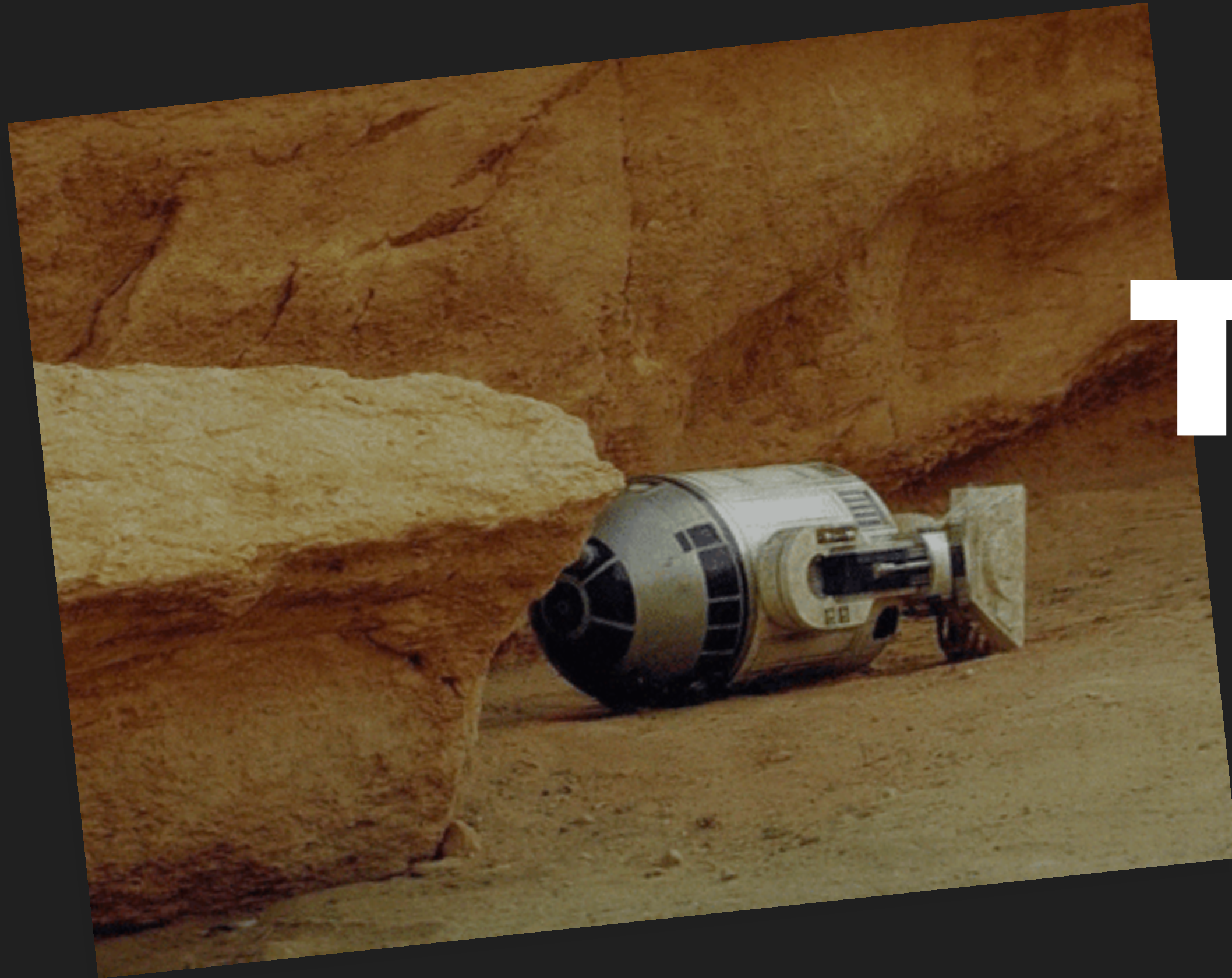
The New York Times











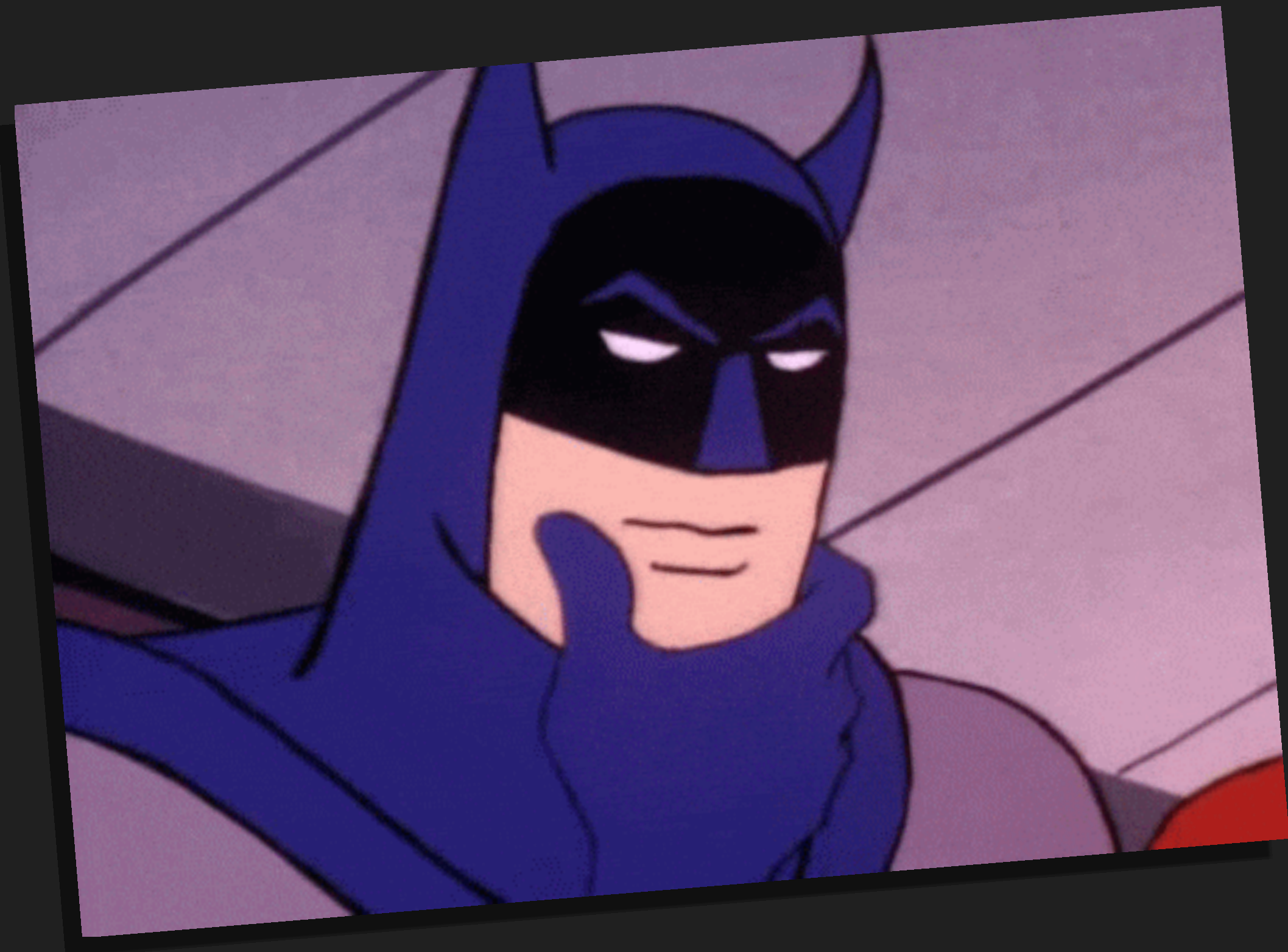
Tired:

**GIF, an image
file format**

Wired:

GLF, a shorthand for silent,
inline animated clips





```

```


Animated

Silent

Looping

Inline

Autoplay

- Animated**
- Silent**
- Looping**
- Inline**
- Autoplay**

Animated

Silent

Looping

Inline

Autoplay

Text alternative

Animated

Silent

Looping

Inline

Autoplay

Text alternative



Of the top 1,000,000
home pages...

Source: webaim.org/projects/million (March 2024)

Of the top 1,000,000
home pages...



Source: webaim.org/projects/million (March 2024)

Of the top 1,000,000
home pages...



54.5%
were missing
alt text!

Source: webaim.org/projects/million (March 2024)



Describe the image for blind and low-vision users

A peeved groundhog repeatedly shouts “Hey!...Hey!...Hey!”



Accessibility

Write Alt Text Like You're Talking To A Friend



[Scott Vandehey](#)

April 22nd, 2024




```

```


Animated

Silent

Looping

Inline

Autoplay

Text alternative



Animated

Silent

Looping

Inline

Autoplay

Text alternative

Higher quality

Animated

Silent

Looping

Inline

Autoplay

Text alternative

Higher quality

Smaller size

Animated

Silent

Looping

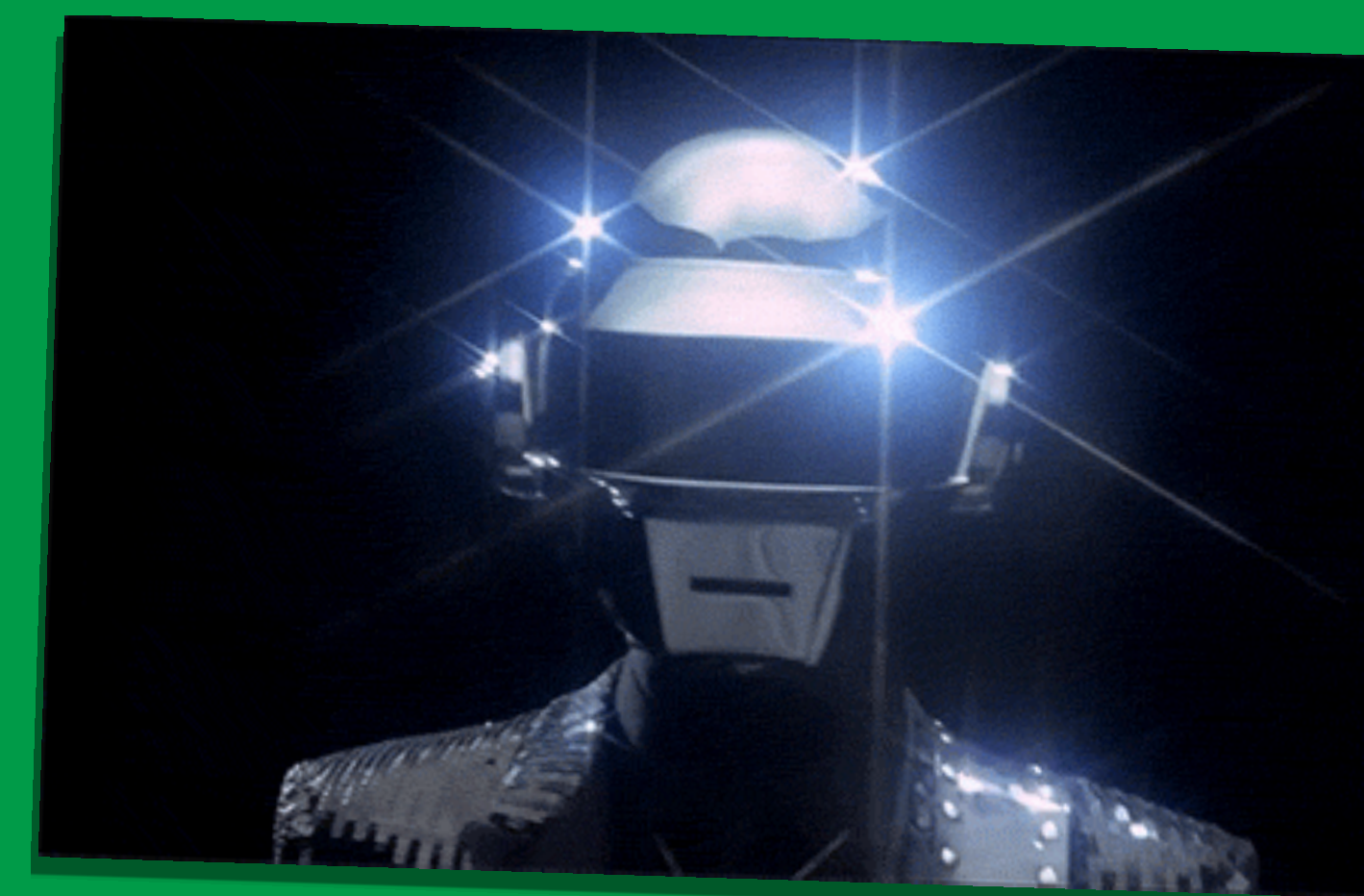
Inline

Autoplay

Text alternative

Higher quality

Smaller size





WebP

Since 2014
(v32)

Since 2018
(v18)

Since 2019
(v65)

Since 2022
(v16.0)



WebP

Since 2014
(v32)

Since 2018
(v18)

Since 2019
(v65)

Since 2022
(v16.0)

AVIF

Since 2020
(v85)

This Year
(v121)

Since 2021
(v93)

Since 2023
(v16.4)

974 KB



GIF



974 KB



GIF

HIGHER
QUALITY!!



226 KB



WebP

66 KB



AVIF

Local

libvips

sharp

gif2webp

FFmpeg

ImageMagick

Local

libvips

sharp

gif2webp

FFmpeg

ImageMagick

Services

 **Cloudinary**

FULL DISCLOSURE:
THEY'RE A CLIENT!

cloudfour.com/made

...and many others!

```

```



```

```

```
<picture>
  <source type="image/avif" srcset="clip.avif" />
  <source type="image/webp" srcset="clip.webp" />
  
</picture>
```


Animated

Silent

Looping

Inline

Autoplay

Text alternative

Higher quality

Smaller size



Animated

Silent

Looping

Inline

Autoplay

Text alternative

Higher quality

Smaller size

Motion prefs

Animated

Silent

Text alternative

Higher quality

Smaller size

Motion prefs



clip.avif

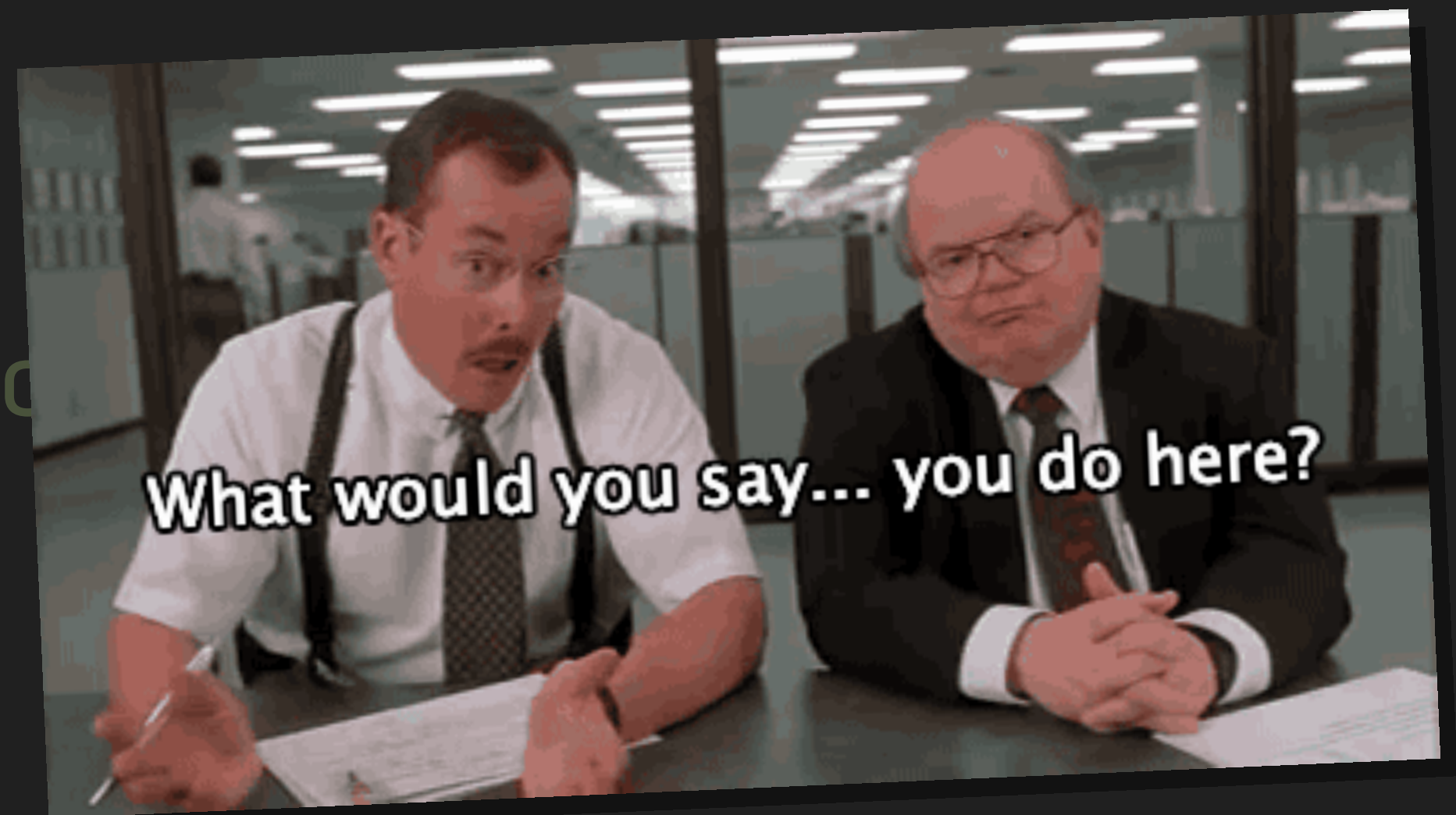


static.avif




```
<picture>
  <source
    media="(prefers-reduced-motion: no-preference)"
    srcset="clip.avif" />
  
</picture>
```

```
<picture>
  <source
    media="(prefers-reduced-mo
    srcset="clip.avif" />
  
</picture>
```





```
<picture>
  <source
    media="(prefers-reduced-motion: no-preference)"
    srcset="clip.avif" />
  
</picture>
```

Animated

Silent

Looping

Inline

Autoplay

Text alternative

Higher quality

Smaller size

Motion prefs



Animated

Silent

Looping

Inline

Autoplay

Text alternative

Higher quality

Smaller size

Motion prefs

Playback control

Animated

Silent

Looping

Inline

Autoplay

Text alter

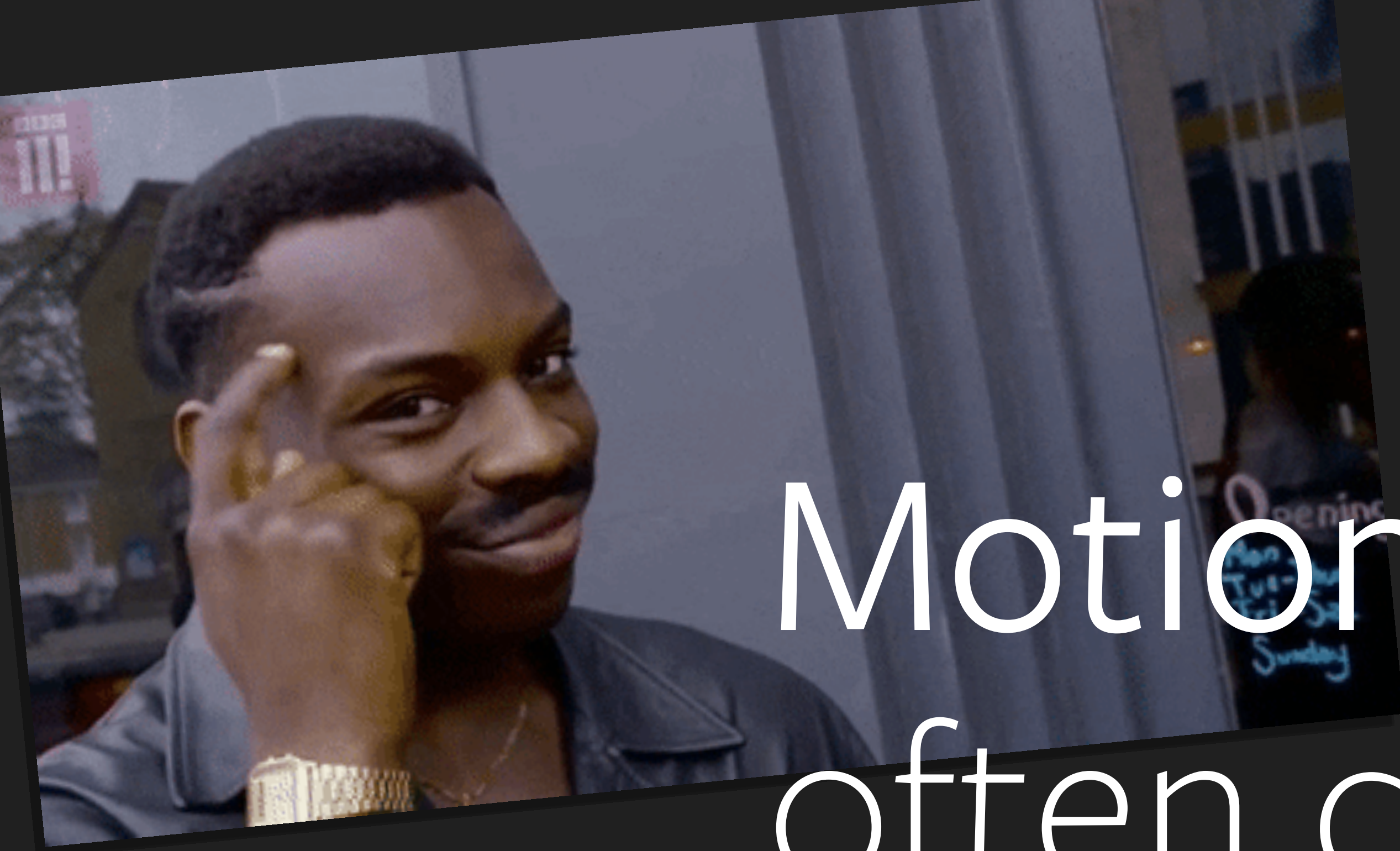
Higher q

Smaller

Motion prefs

Playback control





Motion sensitivity
often depends on
content + context





#







#



<video controls>





MP4
(H.264)

Since 2010
(v4)

Since 2015
(v12)

Since 2015
(v35)

Since 2008
(v3.2)



MP4
(H.264)

Since 2010
(v4)

Since 2015
(v12)

Since 2015
(v35)

Since 2008
(v3.2)

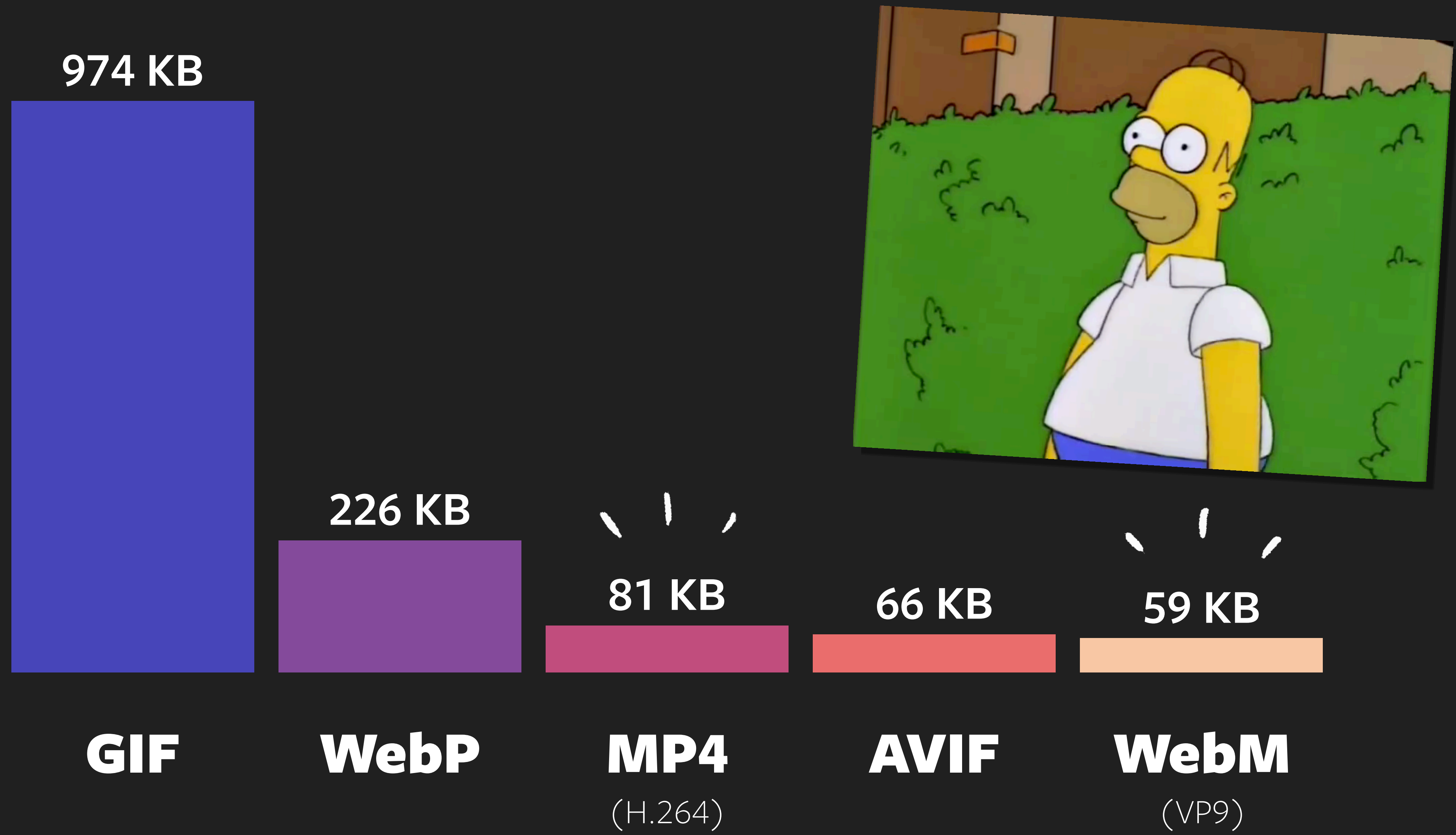
WebM
(VP8 / VP9)

Since 2013
(v25)

Since 2020
(v79)

Since 2014
(v28)

Since 2022
(v16.0)



974 KB

226 KB

81 KB

66 KB

59 KB

GIF

WebP

MP4

(H.264)

AVIF

WebM

(VP9)

Local

FFmpeg

Handbrake

**Adobe Media
Encoder**

Local

FFmpeg

Handbrake

**Adobe Media
Encoder**

Services

 **Cloudinary**

FULL DISCLOSURE:
I'M STILL BIASED!



cloudfour.com/made

...and even more alternatives!

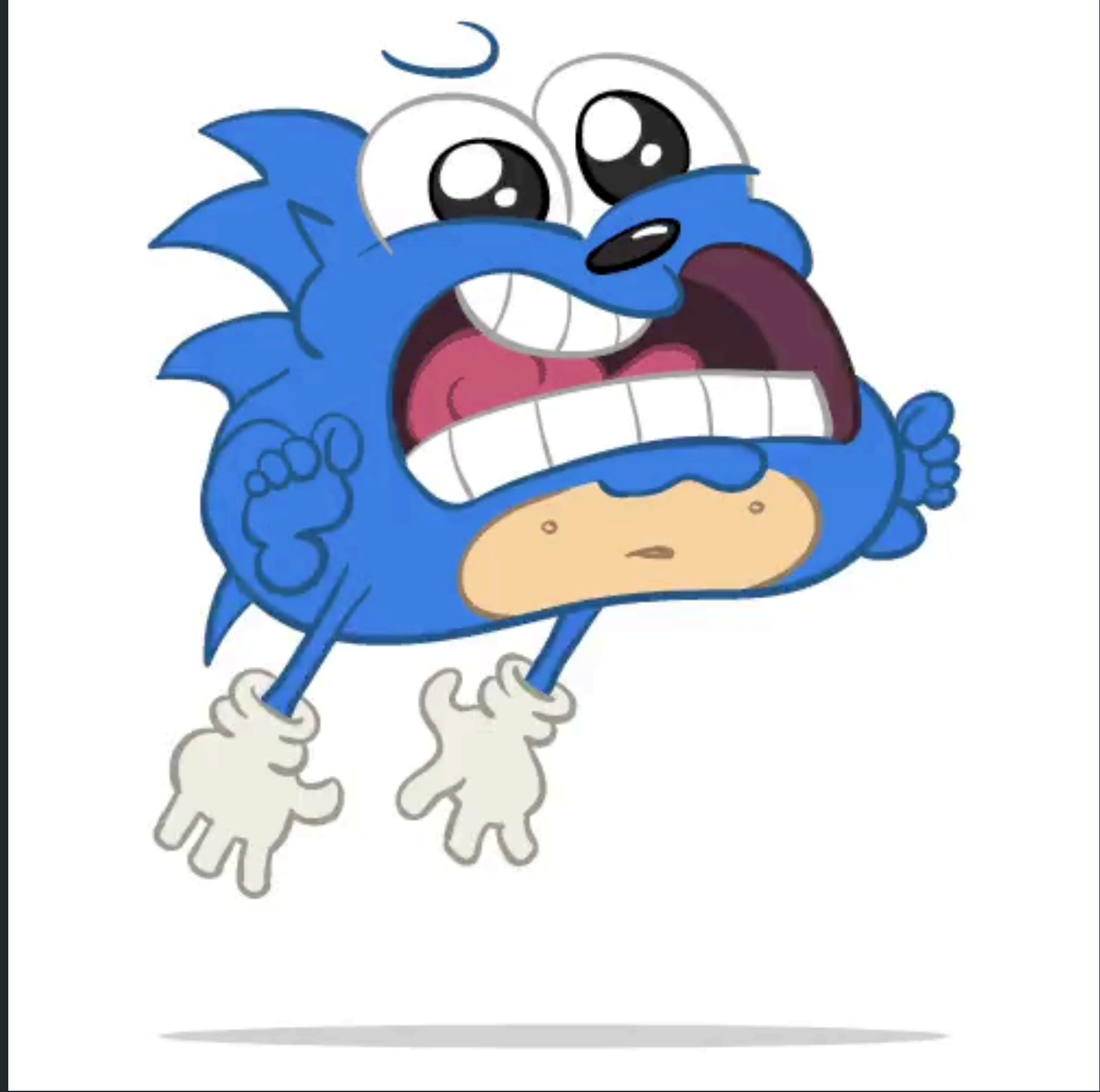
```
<video controls autoplay loop muted playsinline  
  src="clip.mp4"  
  width="..." height="...">  
</video>
```

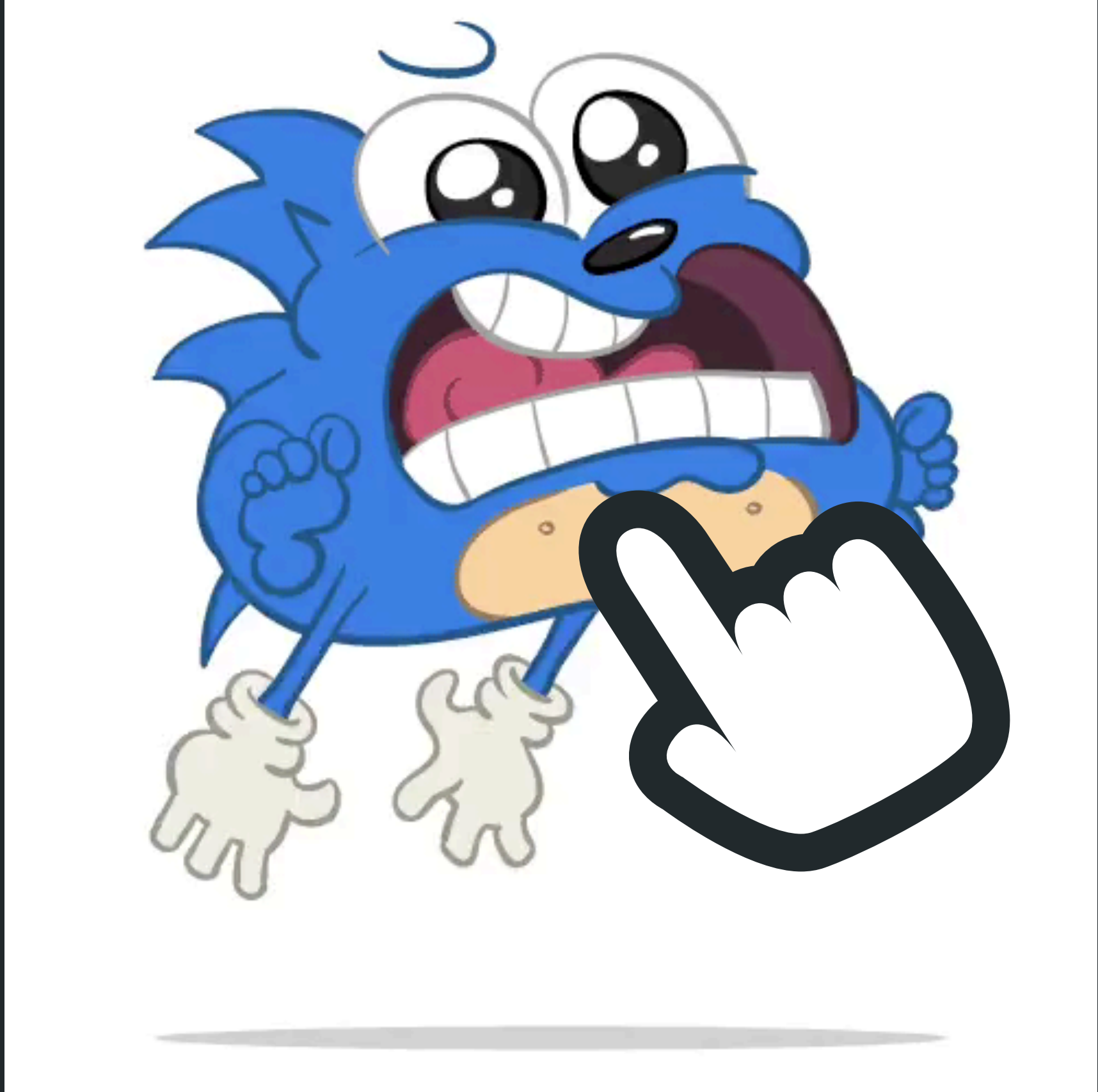


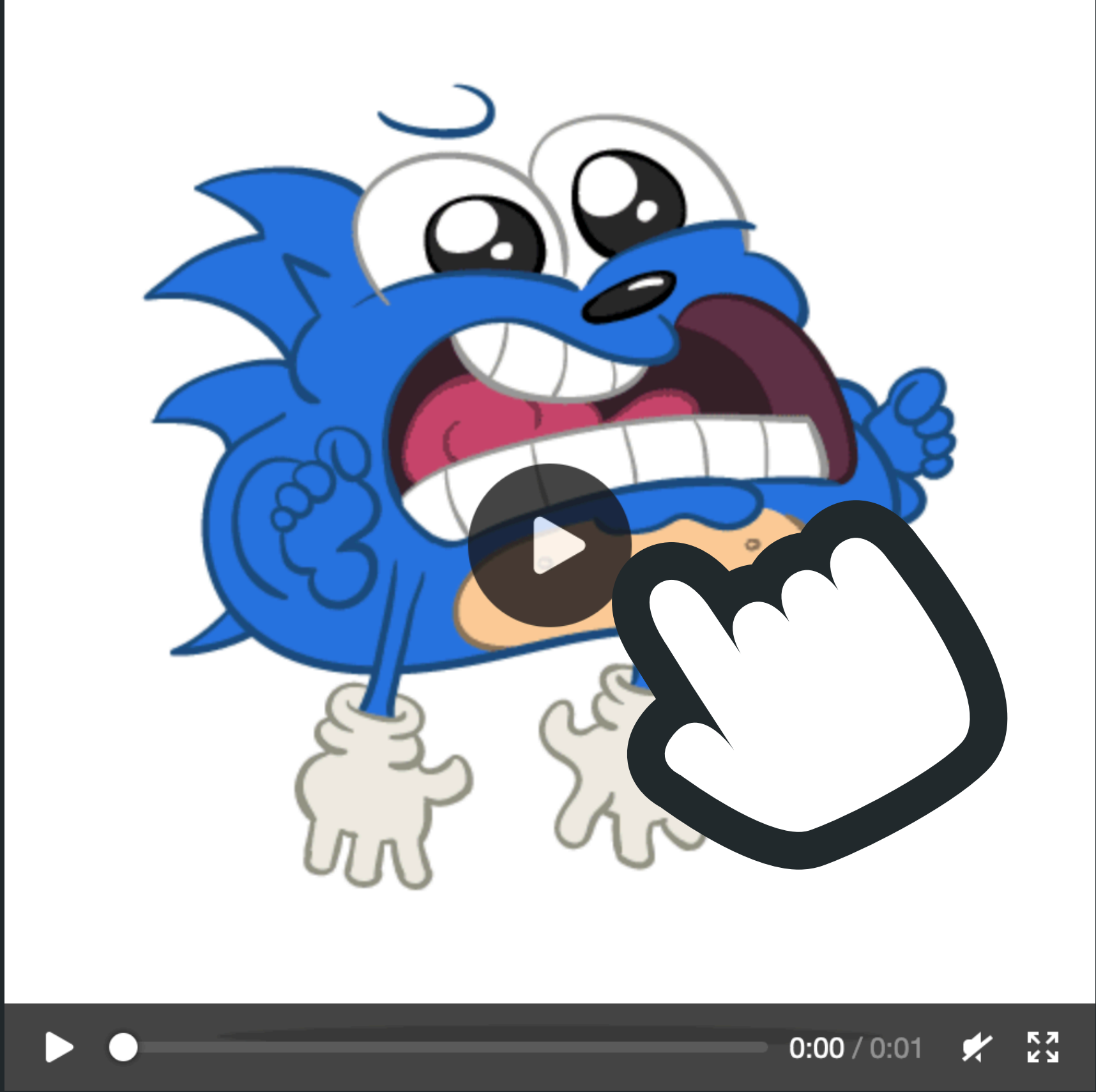
```
<video controls autoplay loop muted playsinline  
  src="clip.mp4"  
  width="..." height="...">  
</video>
```



```
<video controls autoplay loop muted playsinline
  width="..." height="...">
  <source src="clip.webm" type="video/webm" />
  <source src="clip.mp4" type="video/mp4" />
</video>
```









▶ ● 0:00 / 0:01 🔊 🗄

Animated

Silent

Looping

Inline

Autoplay

Text alternative

Higher quality

Smaller size

Motion prefs

Playback control

Animated

Silent

Text alternative

Higher quality

Smaller size

Motion prefs

Playback control



ng

ay


```

```

```

```

```
<video controls autoplay loop muted playsinline  
  src="clip.mp4"  
  width="..." height="...">  
</video>
```

```
<video controls autoplay loop muted playsinline  
  src="clip.mp4"  
  width="..." height="...">  
</video>
```



```
<video controls autoplay loop m
  src="clip.mp4"
  width="..." height="...">
</video>
```



Animated

Silent

Looping

Inline

Autoplay

Text alternative

Higher quality

Smaller size

Motion prefs

Playback control

Animated

Silent

Looping

Inline

Autoplay

~~Text alternative~~

Higher quality

Smaller size

~~Motion prefs~~

Playback control

✓ Animated

■ ~~Text alternative~~

✓

✓

✓

✓ Autoplay

✓ Playback control



```
<video controls autoplay loop muted playsinline  
  width="..." height="...">  
  <source src="clip.webm" type="video/webm" />  
  <source src="clip.mp4" type="video/mp4" />  
</video>
```

```
<figure>
  <video controls autoplay loop muted playsinline
    width="..." height="...">
    <source src="clip.webm" type="video/webm" />
    <source src="clip.mp4" type="video/mp4" />
  </video>
  <figcaption>
    An example of a video used as a GIF.
  </figcaption>
</figure>
```



```
<video controls autoplay loop muted playsinline
  width="..." height="..."
  aria-label="An example of a video used as a GIF.">
  <source src="clip.webm" type="video/webm" />
  <source src="clip.mp4" type="video/mp4" />
</video>
```

```
<video controls autoplay loop muted playsinline
  width="..." height="..."
  aria-labelledby="clip-label">
  <source src="clip.webm" type="video/webm" />
  <source src="clip.mp4" type="video/mp4" />
</video>
<div id="clip-label" aria-hidden="true">
  An example of a video used as a GIF.
</div>
```

Animated

Silent

Looping

Inline

Autoplay

~~Text alternative~~

Higher quality

Smaller size

~~Motion prefs~~

Playback control

Animated

Silent

Looping

Inline

Autoplay

Text alternative

Higher quality

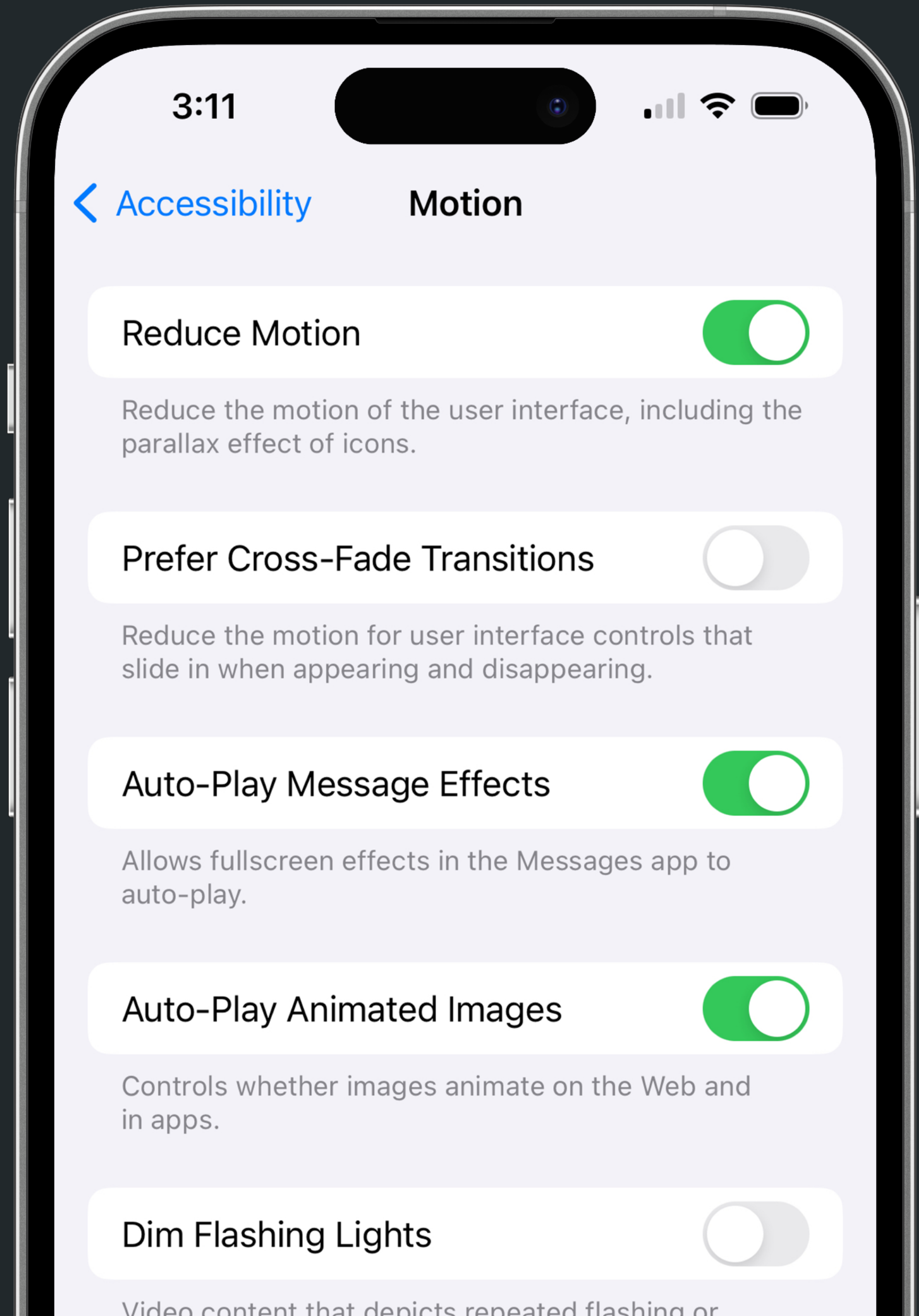
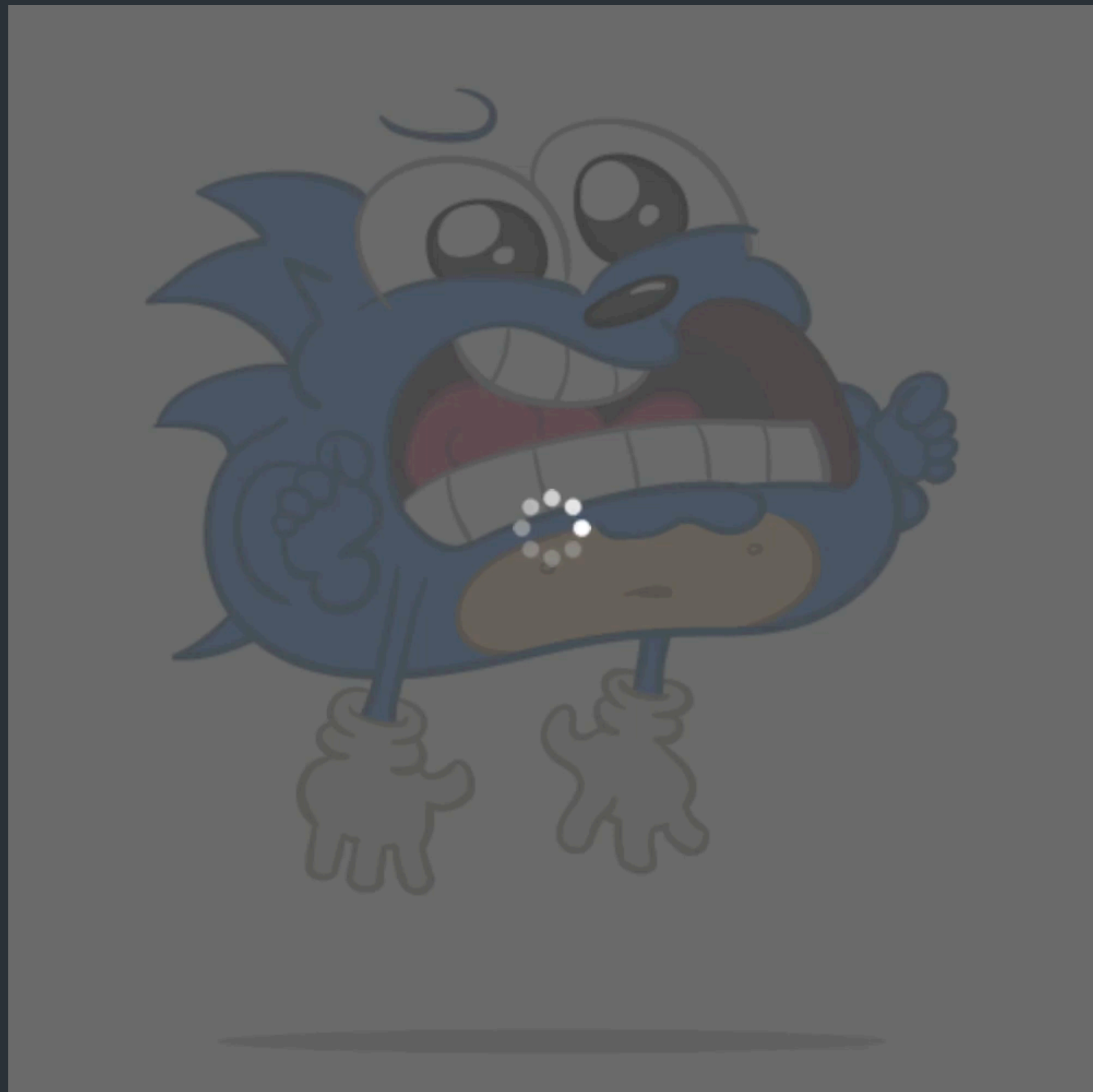
Smaller size

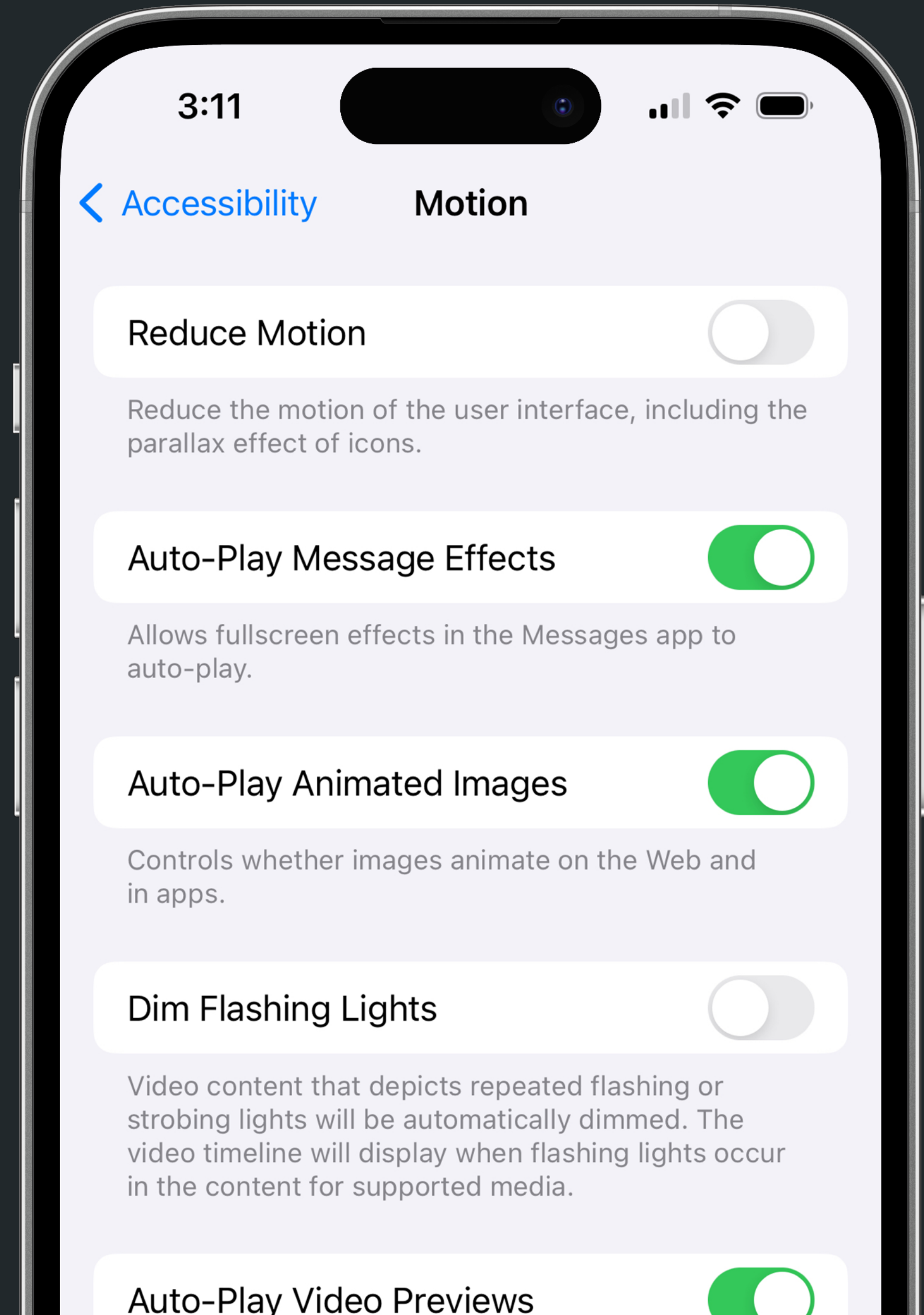
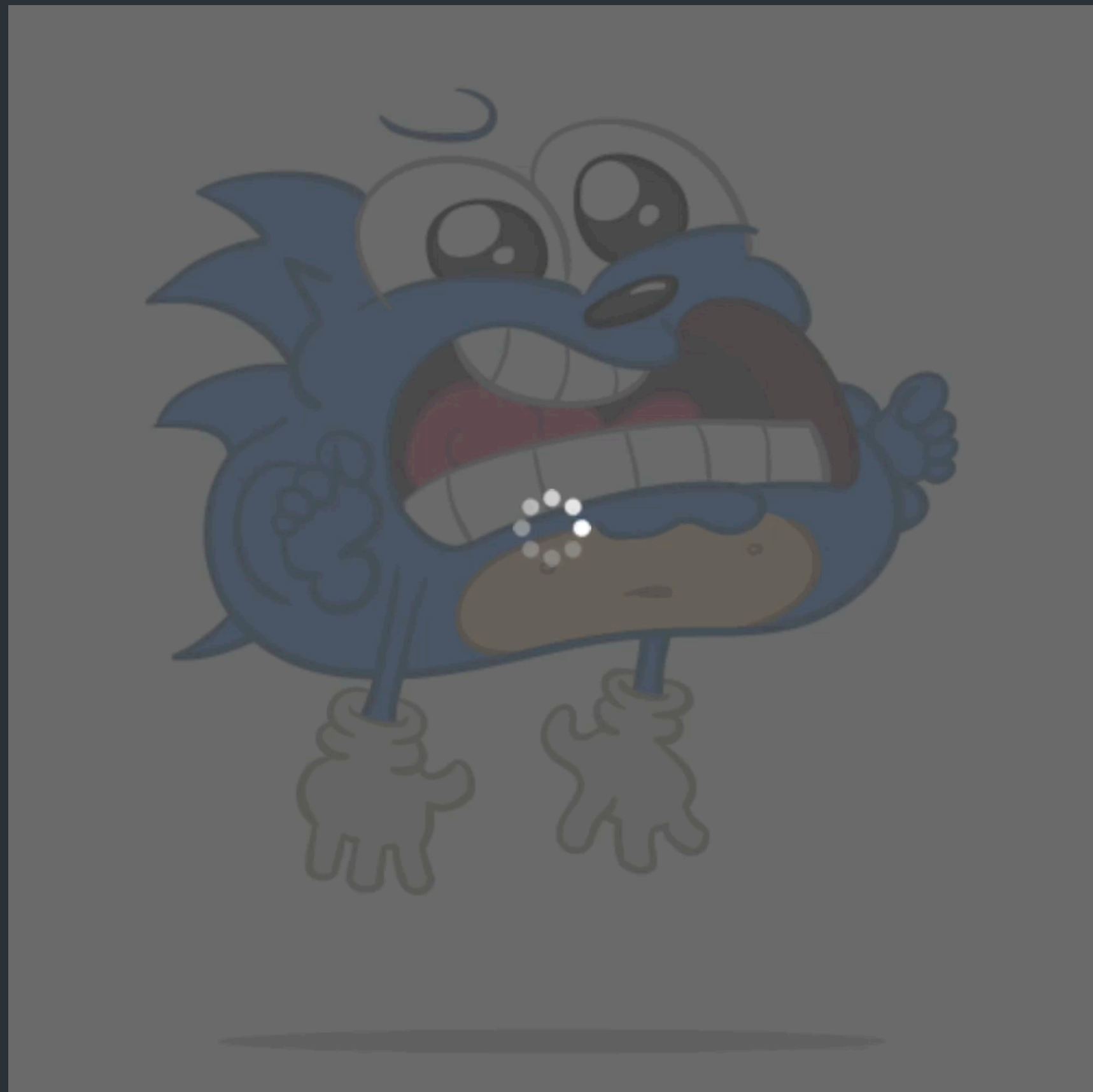
~~**Motion prefs**~~

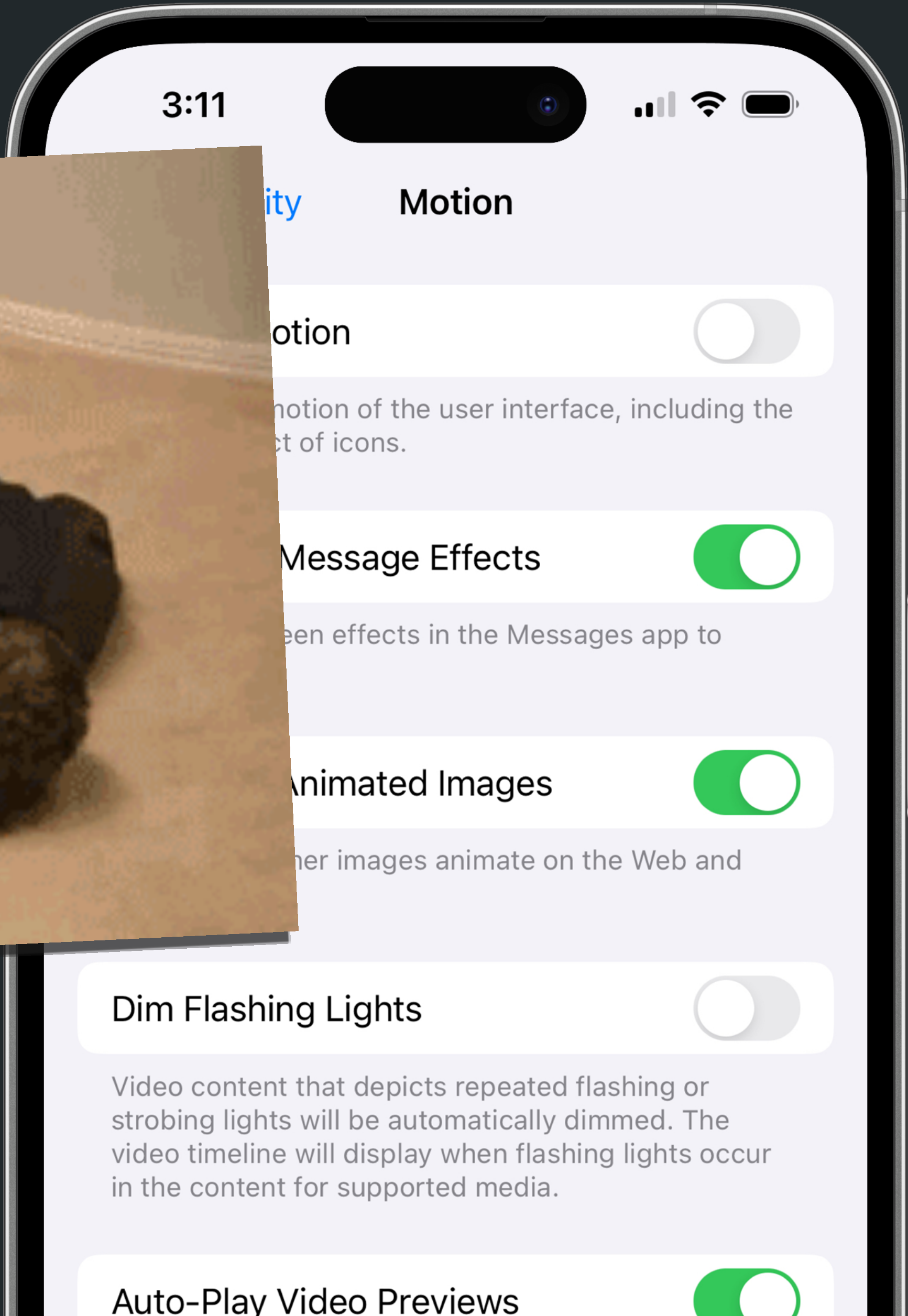
Playback control

```
<picture>
  <source
    media="(prefers-reduced-motion: no-preference)"
    srcset="clip.avif" />
  
</picture>
```

```
<video controls autoplay loop muted playsinline
  width="..." height="..."
  aria-labelledby="clip-label">
  <source src="clip.webm" type="video/webm"
    media="(prefers-reduced-motion: no-preference)" />
  <source src="clip.mp4" type="video/mp4"
    media="(prefers-reduced-motion: no-preference)" />
</video>
<div id="clip-label" aria-hidden="true">
  An example of a video used as a GIF.
</div>
```





A low-resolution, pixelated image of Superman standing with his hands on his hips. He is wearing his classic blue suit with a red and yellow 'S' shield on his chest. The background is a blue sky with a white diagonal streak. The text 'Web components!' is overlaid in the center in a bold, white, sans-serif font.

Web components!


```
<video controls autoplay loop muted playsinline
  width="..." height="..."
  aria-labelledby="clip-label">
  <source src="clip.webm" type="video/webm" />
  <source src="clip.mp4" type="video/mp4" />
</video>
<div id="clip-label" aria-hidden="true">
  An example of a video used as a GIF.
</div>
```

```
<video controls loop muted playsinline
  width="..." height="..."
  aria-labelledby="clip-label">
  <source src="clip.webm" type="video/webm" />
  <source src="clip.mp4" type="video/mp4" />
</video>
<div id="clip-label" aria-hidden="true">
  An example of a video used as a GIF.
</div>
```



```
class GifLike extends HTMLElement {
  static motionQuery = window.matchMedia(
    "(prefers-reduced-motion: no-preference)"
  );

  connectedCallback() {
    this.video = this.querySelector("video");

    GifLike.motionQuery.addEventListener("change", (query) => {
      this.toggle(query.matches);
    });

    this.toggle(GifLike.motionQuery.matches);
  }

  toggle(state) {
    if (state) {
      this.video.play();
    } else {
      this.video.pause();
    }
  }
}

customElements.define("gif-like", GifLike);
```

```
class GifLike extends HTMLElement {
  static motionQuery = window.matchMedia(
    "(prefers-reduced-motion: no-preference)"
  );

  connectedCallback() {
    this.video = this.querySelector("video");

    GifLike.motionQuery.addEventListener("change", (query) => {
      this.toggle(query.matches);
    });

    this.toggle(GifLike.motionQuery.matches);
  }

  toggle(state) {
```



```
class GifLike extends HTMLElement {
  static motionQuery = window.matchMedia(
    "(prefers-reduced-motion: no-preference)"
  );

  connectedCallback() {
    this.video = this.querySelector("video");

    GifLike.motionQuery.addEventListener("change", (query) => {
      this.toggle(query.matches);
    });

    this.toggle(GifLike.motionQuery.matches);
  }

  toggle(state) {
```



```
class GifLike extends HTMLElement {
  static motionQuery = window.matchMedia(
    "(prefers-reduced-motion: no-preference)"
  );

  connectedCallback() {
    this.video = this.querySelector("video");

    GifLike.motionQuery.addEventListener("change", (query) => {
      this.toggle(query.matches);
    });

    this.toggle(GifLike.motionQuery.matches);
  }

  toggle(state) {
```



```
class GifLike extends HTMLElement {
  static motionQuery = window.matchMedia(
    "(prefers-reduced-motion: no-preference)"
  );

  connectedCallback() {
    this.video = this.querySelector("video");

    GifLike.motionQuery.addEventListener("change", (query) => {
      this.toggle(query.matches);
    });

    this.toggle(GifLike.motionQuery.matches);
  }

  toggle(state) {
```

```
class GifLike extends HTMLElement {
  static motionQuery = window.matchMedia(
    "(prefers-reduced-motion: no-preference)"
  );

  connectedCallback() {
    this.video = this.querySelector("video");

    GifLike.motionQuery.addEventListener("change", (query) => {
      this.toggle(query.matches);
    });

    this.toggle(GifLike.motionQuery.matches);
  }

  toggle(state) {
```

```
});
```

```
  this.toggle(GifLike.motionQuery.matches);
```

```
}
```

```
toggle(state) {
```

```
  if (state) {
```

```
    this.video.play();
```

```
  } else {
```

```
    this.video.pause();
```

```
  }
```

```
}
```

```
}
```

```
customElements.define("gif-like", GifLike);
```



```
toggle(state) {  
  if (state) {  
    this.video.play();  
  } else {  
    this.video.pause();  
  }  
}  
}  
}
```

```
customElements.define("gif-like", GifLike);
```

```
class GifLike extends HTMLElement {
  static motionQuery = window.matchMedia(
    "(prefers-reduced-motion: no-preference)"
  );

  connectedCallback() {
    this.video = this.querySelector("video");

    GifLike.motionQuery.addEventListener("change", (query) => {
      this.toggle(query.matches);
    });

    this.toggle(GifLike.motionQuery.matches);
  }

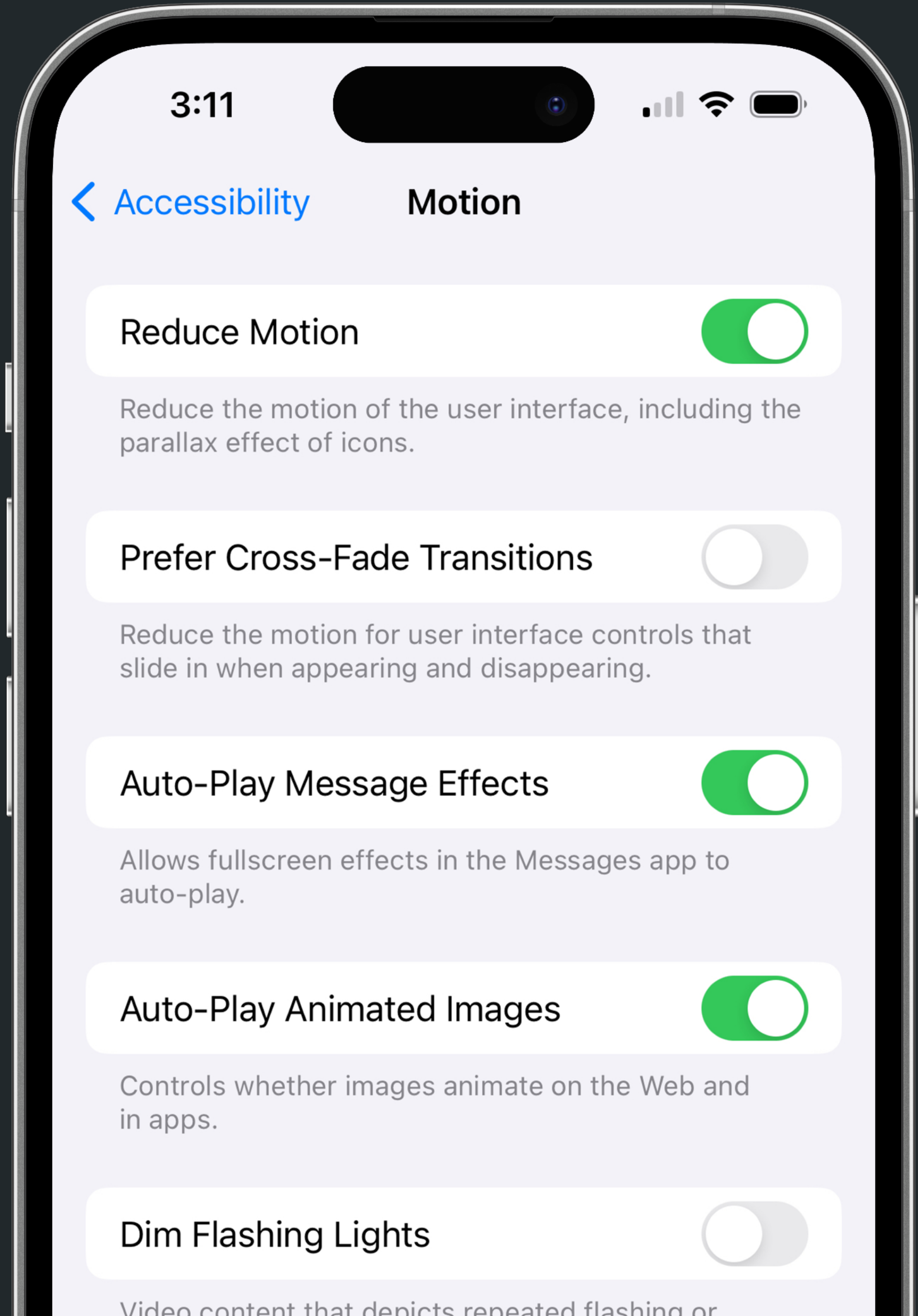
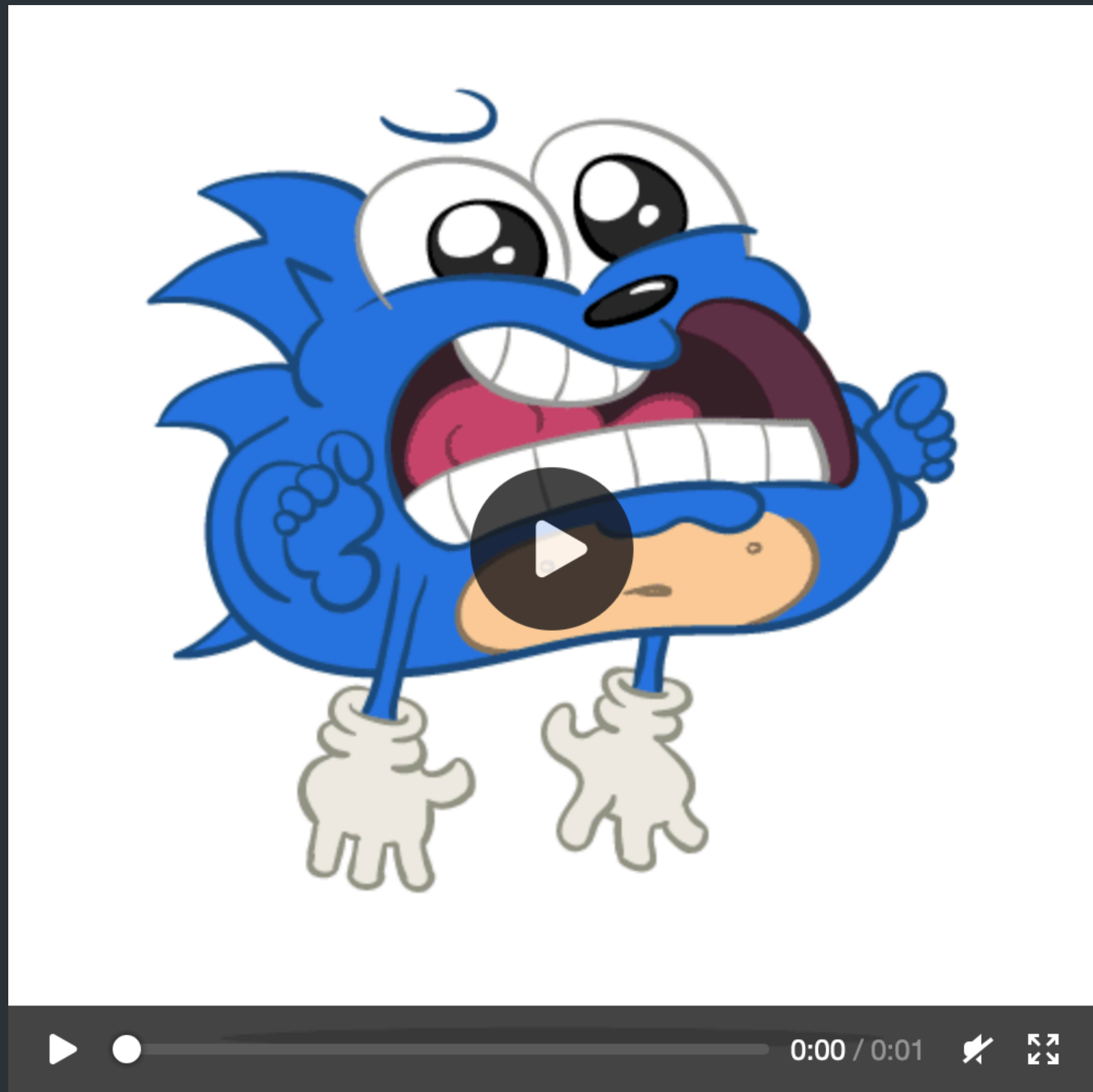
  toggle(state) {
    if (state) {
      this.video.play();
    } else {
      this.video.pause();
    }
  }
}

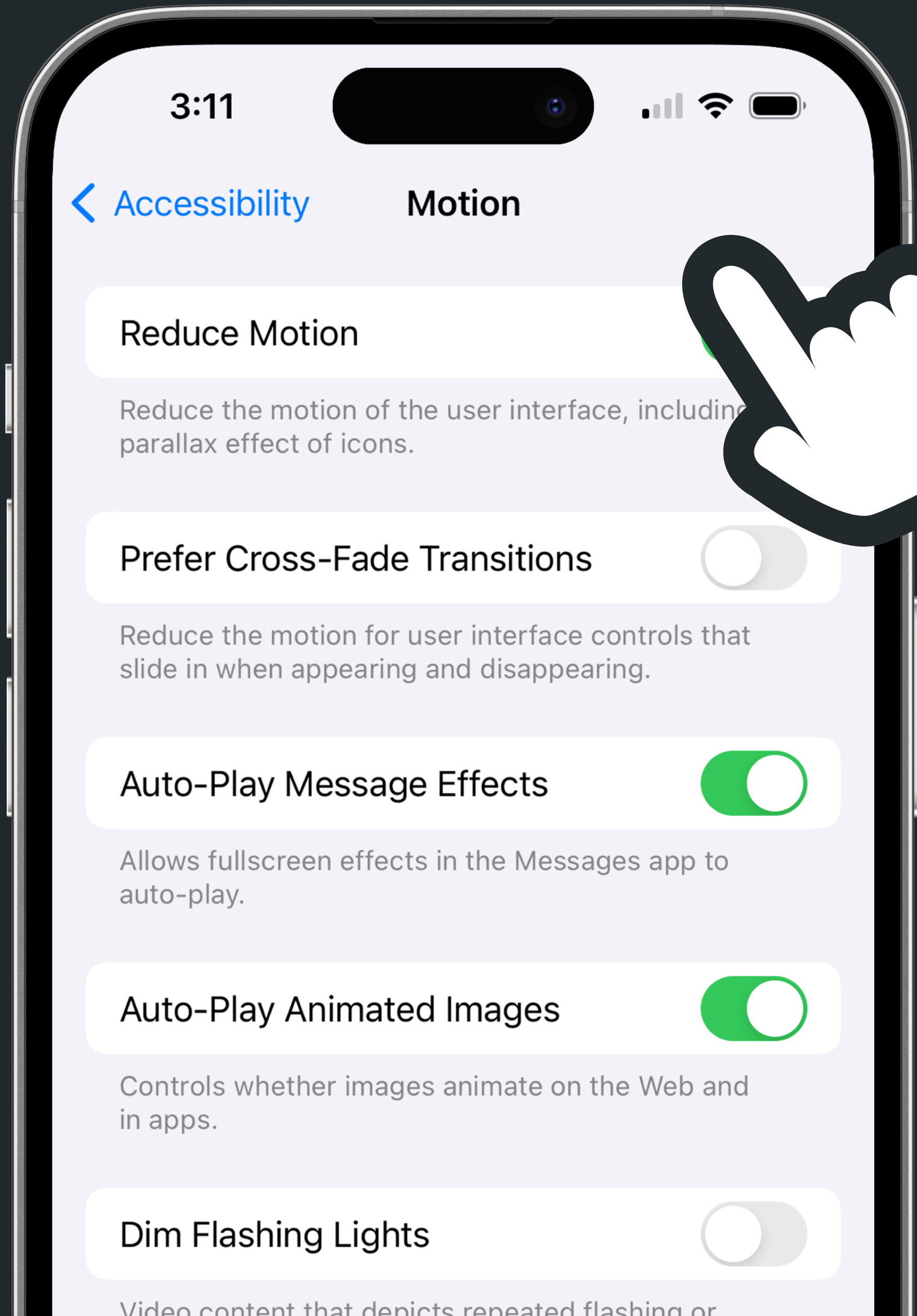
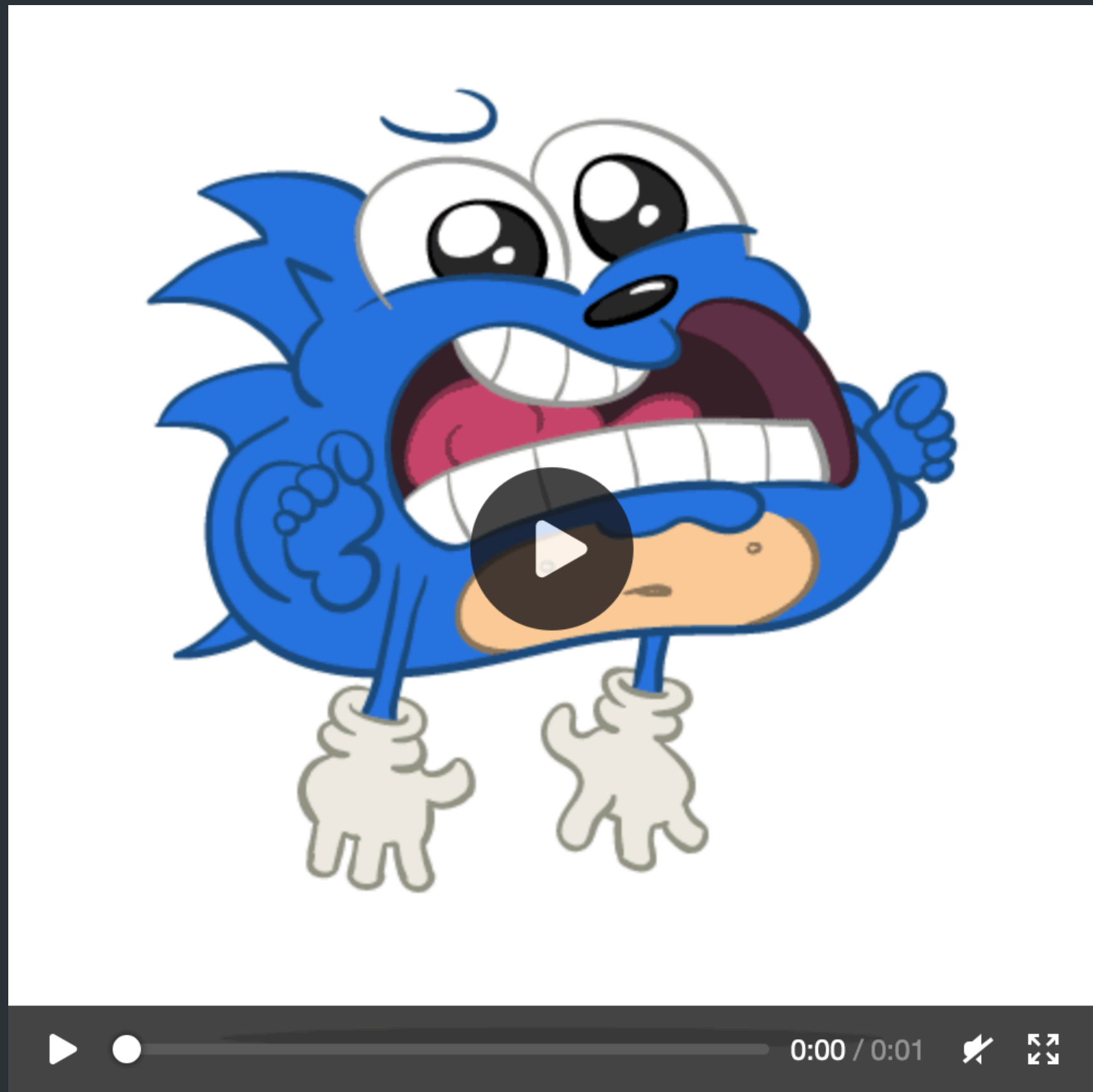
customElements.define("gif-like", GifLike);
```

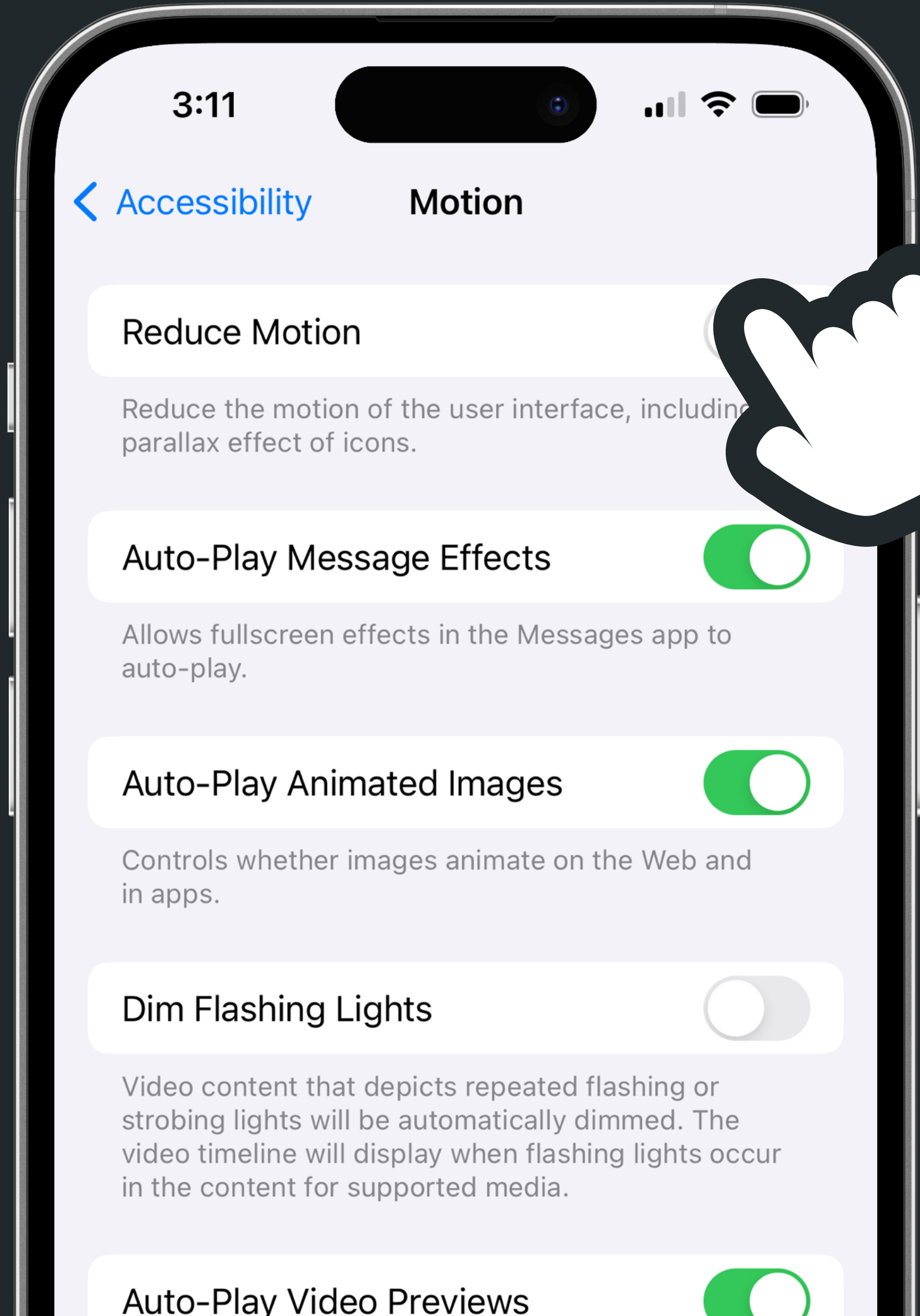
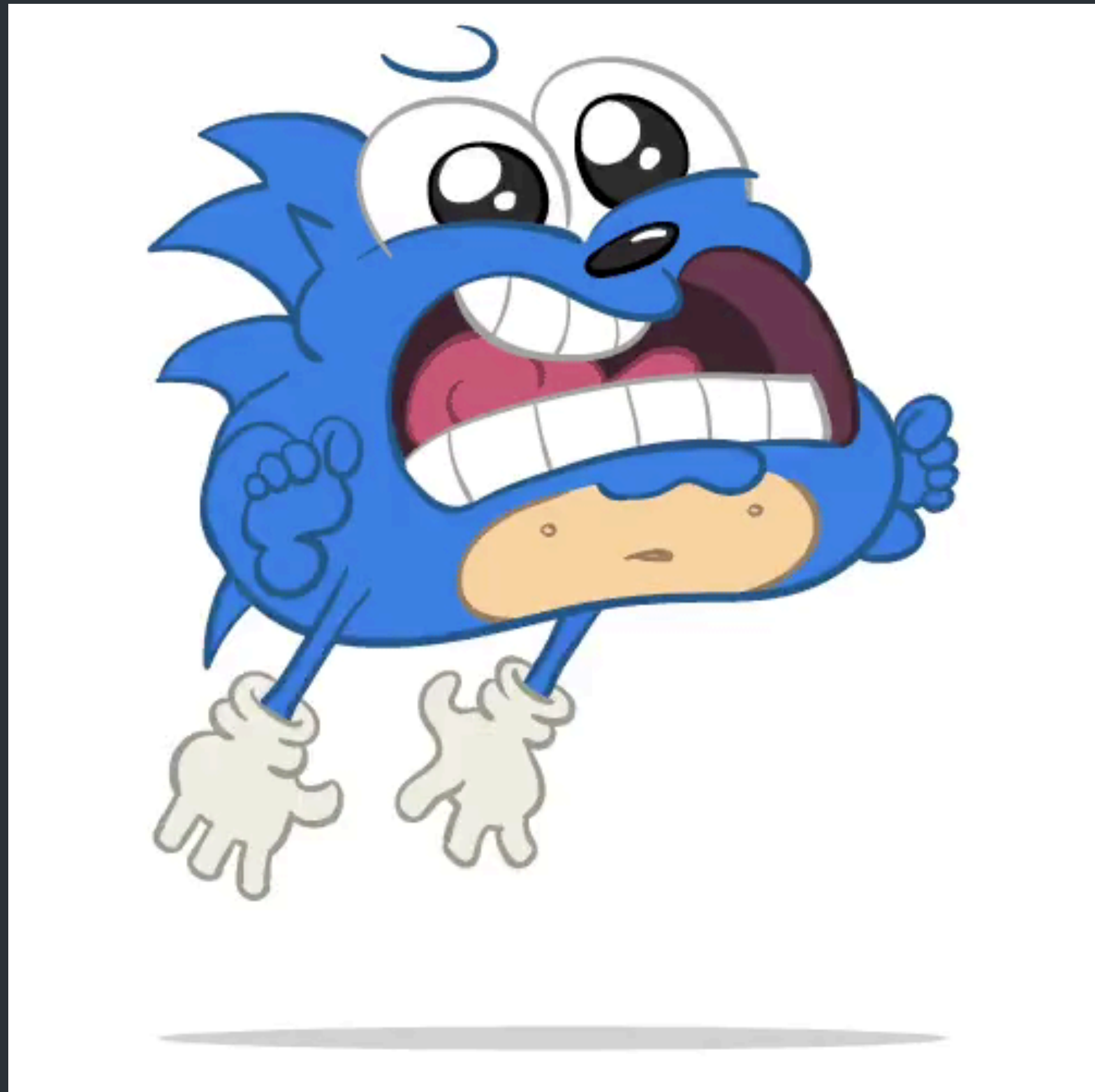
SHARED HERE!

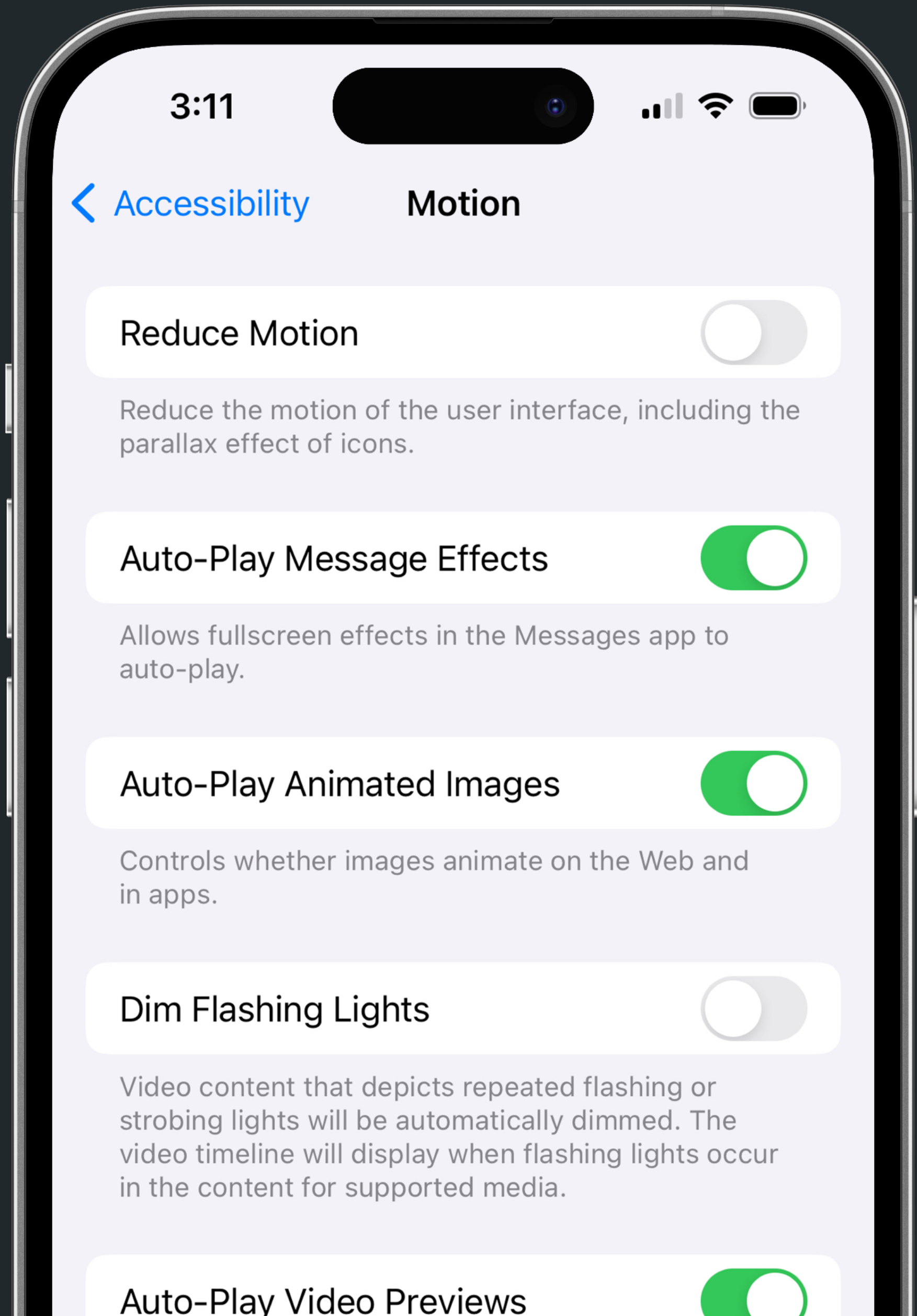
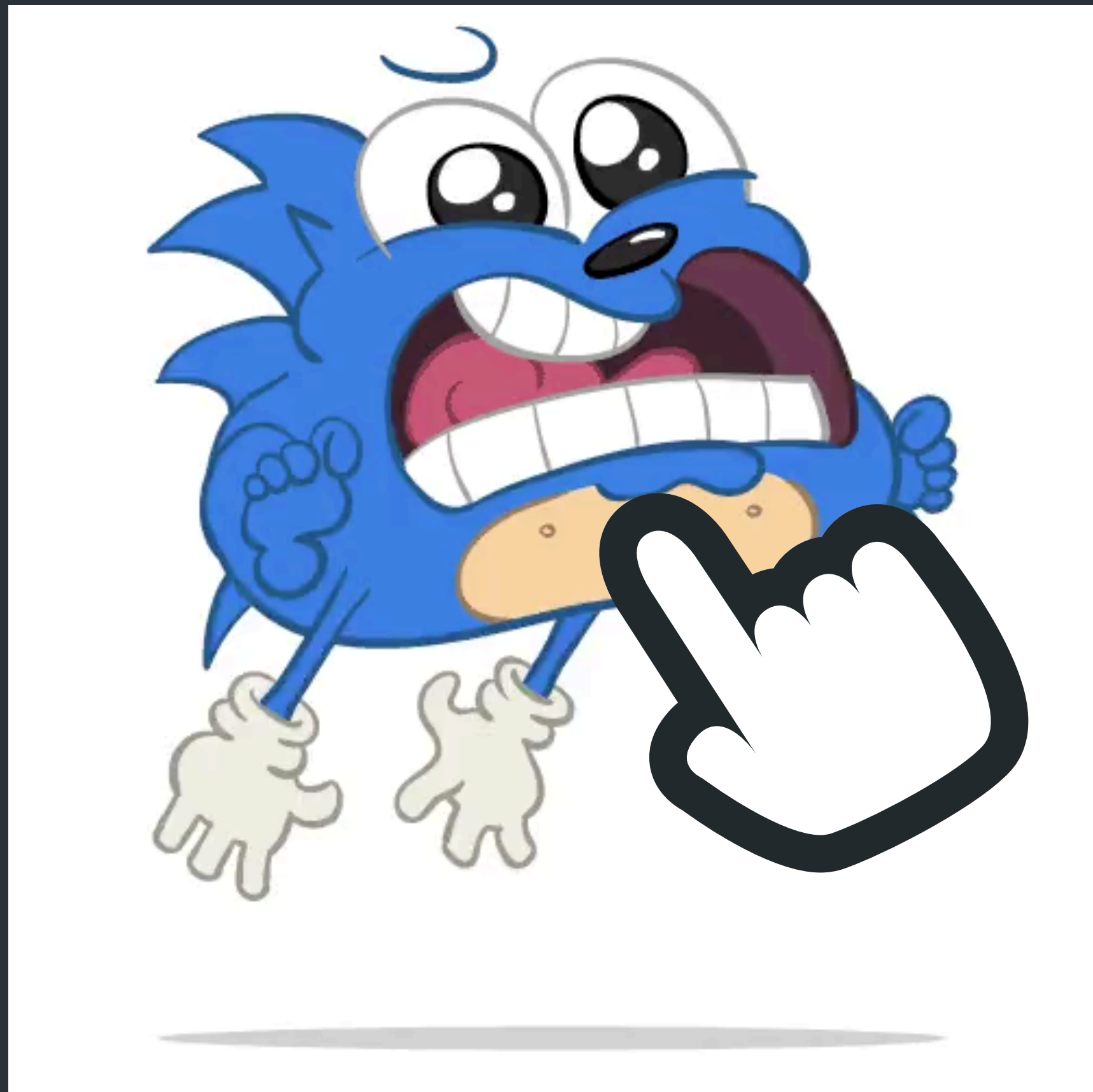


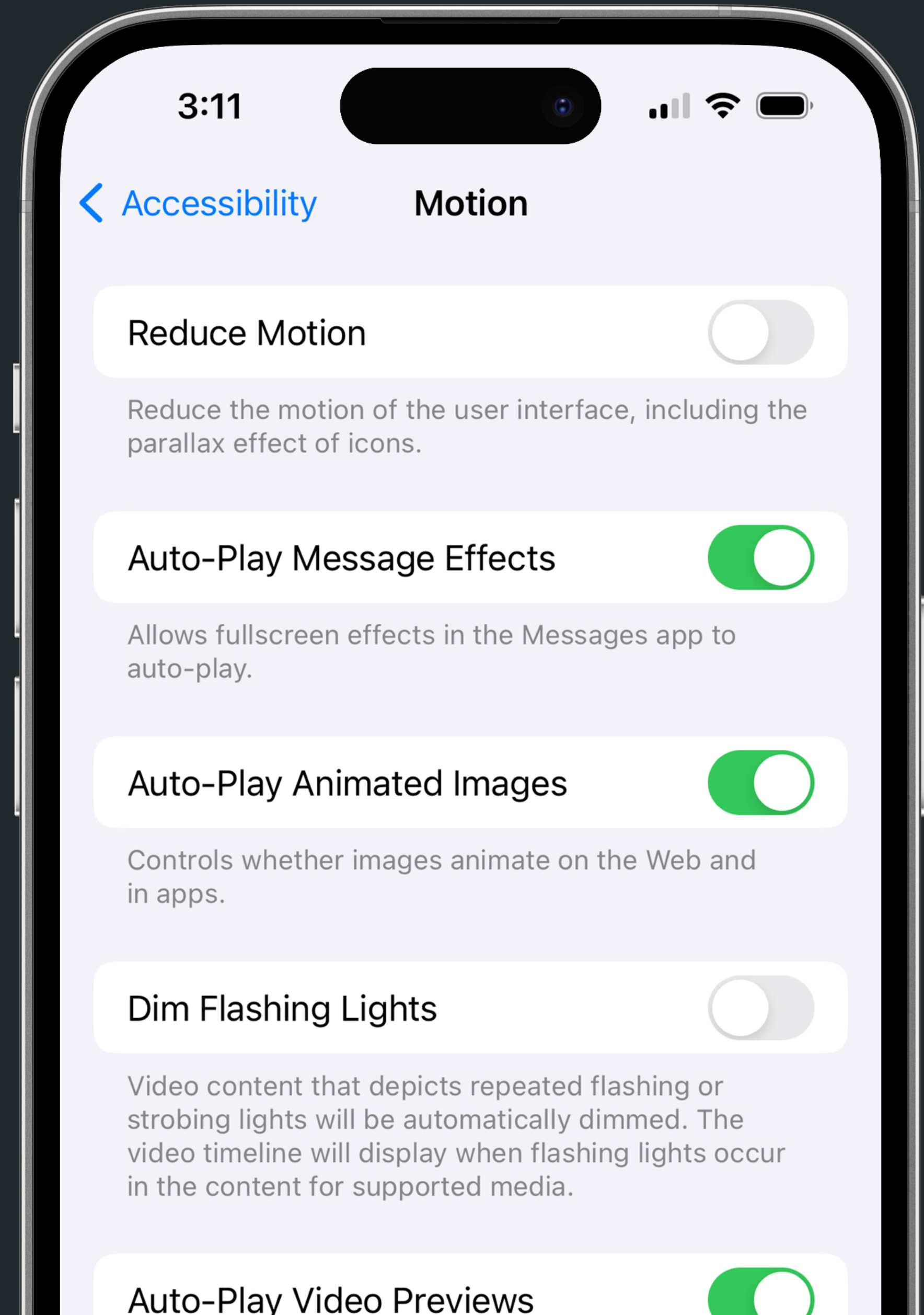
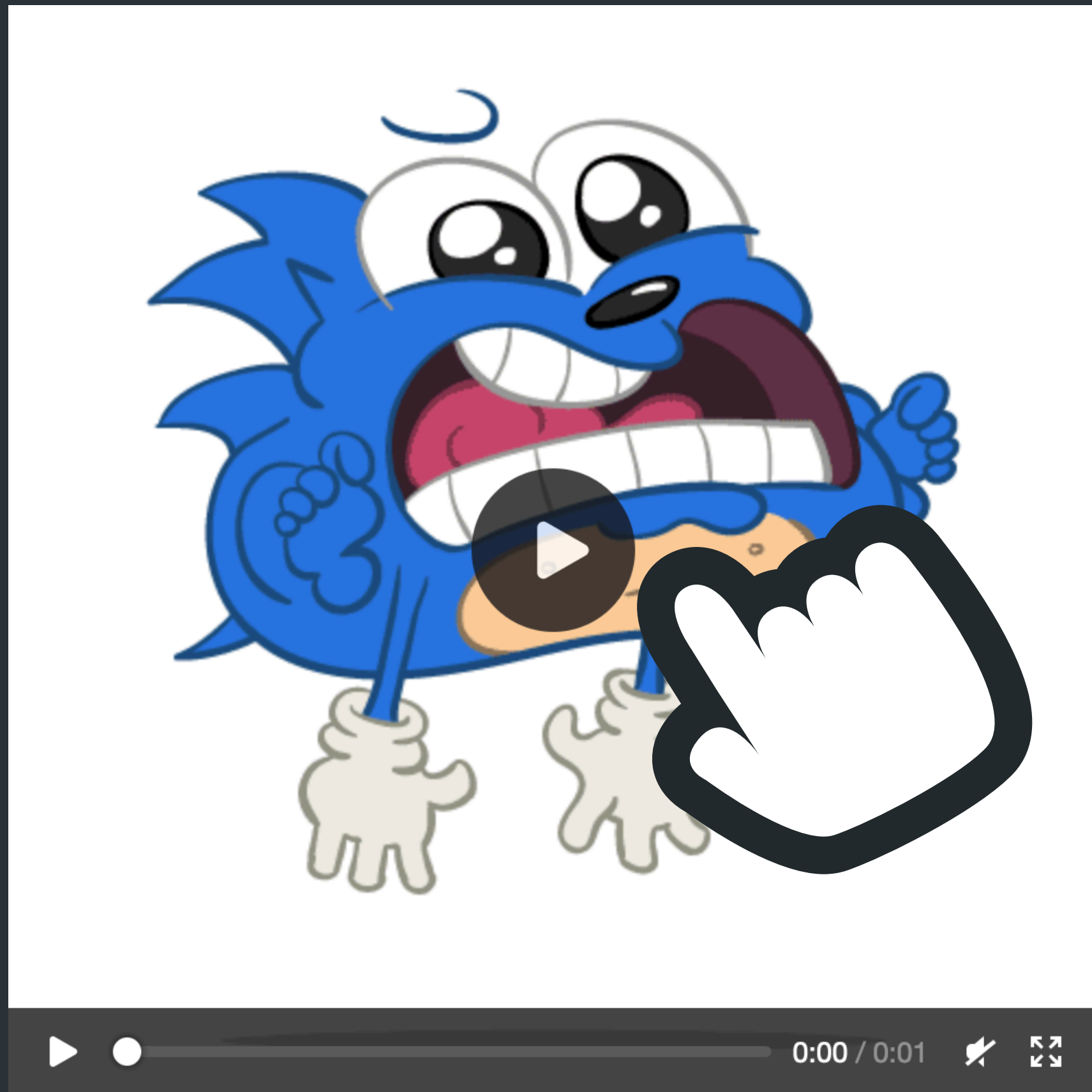
gist.github.com/tylersticka











Animated

Silent

Looping

Inline

Autoplay

Text alternative

Higher quality

Smaller size

~~Motion prefs~~

Playback control

Animated

Silent

Looping

Inline

Autoplay

Text alternative

Higher quality

Smaller size

Motion prefs

Playback control



Am



Si



Lo



In



A



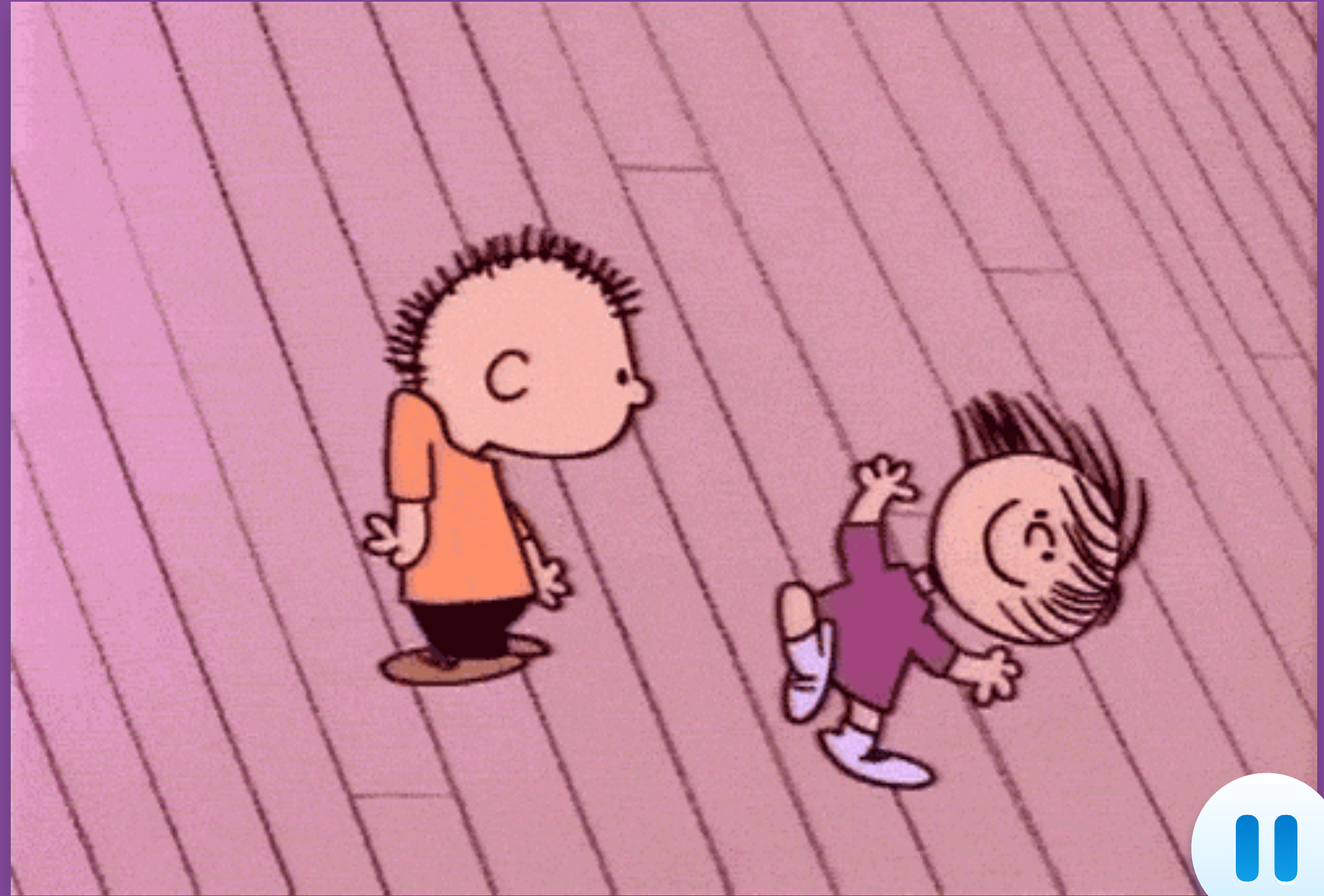
tive

ity

e

fs

ontrol



Collections

Core Web Vitals

Metrics

Fast load times

AI

Optimize Interaction to Next Paint (INP)

Progressive Web Apps

Accessible to all

Network reliability

Safe and secure

Easily discoverable

Web Payments

More

Introduction

Why does speed matter?

What is speed?

How to measure speed?

How to stay fast?

Core Web Vitals

Web Vitals

User-centric performance metrics

Defining the Core Web Vitals metrics thresholds

Largest Contentful Paint (LCP)

Cumulative Layout Shift (CLS)

Interaction to Next Paint (INP)

Optimize Largest Contentful Paint

Optimize Cumulative Layout Shift

Optimize Interaction to Next Paint

[Home](#) > [Articles](#) > [Explore](#) > [Fast load times](#)Was this helpful?  

Lazy loading video



Jeremy Wagner



Rachel Andrew



As with [image elements](#), you can also lazy-load video. Videos are commonly loaded with the `<video>` element (although [an alternate method using](#) has emerged with limited implementation). *How to lazy-load <video>* depends on the use case, though. Let's discuss a couple of scenarios that each require a different solution.

For video that doesn't autoplay

For videos where playback is initiated by the user (that is, videos that *don't* autoplay), specifying the [preload attribute](#) on the `<video>` element may be desirable:

```
<video controls preload="none" poster="one-does-not-simply-placeholder.jpg">
  <source src="one-does-not-simply.webm" type="video/webm">
  <source src="one-does-not-simply.mp4" type="video/mp4">
</video>
```



Key point: A video `poster` image can qualify as an [LCP candidate](#). If your `poster` image is an LCP candidate, you should [preload it](#) with a [fetchpriority attribute](#) value of `"high"` so the user sees it as soon as possible.

Example above uses a `preload` attribute with a value of `none` to prevent browsers from preloading *any* video

The `poster` attribute gives the `<video>` element a placeholder that will occupy the space while the video loads.

Reason for this is that default behaviors for loading video can vary from browser to browser:


```
<gif-like src="..." alt="..."></gif-like>
```



ENHANCE SSR OR ELEVENTY WEBC OR
CMS BLOCK OR TEMPLATE HELPER...



```
<gif-like>  
  <video controls loop muted playsinline  
    preload="none"  
    poster="..."  
    width="..." height="..."  
    aria-labelledby="clip-label">
```

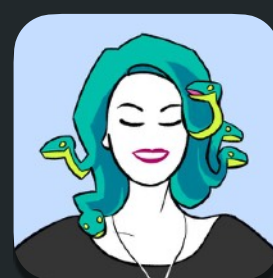



***The possibilities are
pretty much endless.***





Handwritten scribbles in white ink.



Handwritten scribbles in white ink.



GIF is a shorthand for silent, inline, animated clips



Alternative text is
chronically overlooked
low-hanging fruit

974 KB



GIF

Newer formats work in all modern browsers, load faster and look better

226 KB



WebP

81 KB



MP4
(H.264)

66 KB



AVIF

59 KB



WebM
(VP9)

Video gives users
more control

START
STOP

BASIC RUN/STOP

```
<gif-like>
```

Web components are a
great choice for progressive
enhancement

```
</gif-like>
```








tylersticka.com

 @tylersticka@social.lol

 /in/tylersticka

HIRE MY TEAM!



cloudfour.com