

# **DVD Formats and Aspect Ratios Explained**

Shaun Graham – Project Tutorials  
**[PolyMesh.nwcolor.com](http://PolyMesh.nwcolor.com)**

## Introduction

This article is for those like me who continually want higher resolution and like to push the limits of their latest video devices.

Have you ever tried to capture an image from a television device and encountered a squished looking image? Have you ever tried to create a DVD using images from your computer only to find abnormally stretched looking images? Surprisingly, people don't always see the difference, but for those of us that can, it can be really frustrating.

In my search for pixel bliss, I ran into some oddities and issues in regard to getting my digital work onto my home theater system without strange stretching or quality loss. I will attempt to explain some of the things I learned during several years of trial, error, and research on Television aspect ratios and DVD formats.

## A Bit of History

Your monitor and TV are inherently different. Before HDTV, TV's did not have square pixels like monitors do. This is the pixel you are used to seeing on monitors:



(note the square shape)

However older TV's have elongated pixels that are called non-square. They look something like this:

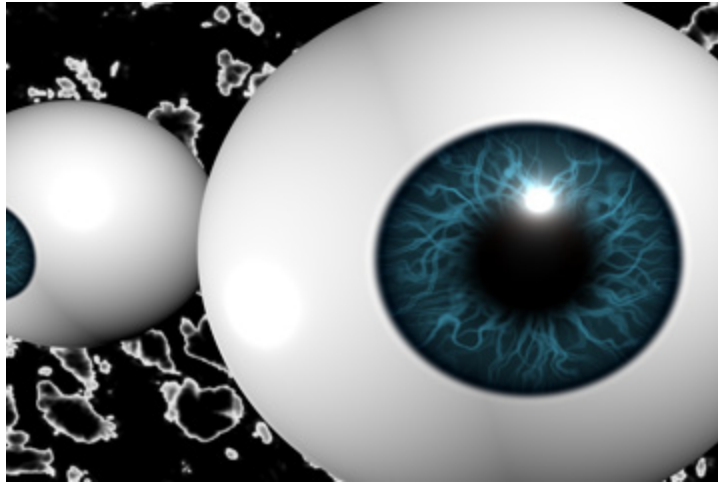


(note the rectangular shape)

Rectangular was ideal for old technology because you could get a much larger viewing area out of fewer pixels, and fewer pixels means less bandwidth requirements. Although the future is square, this old technology will unfortunately be hanging around a bit longer for compatibility sake. We have to deal with that.

## See the Difference

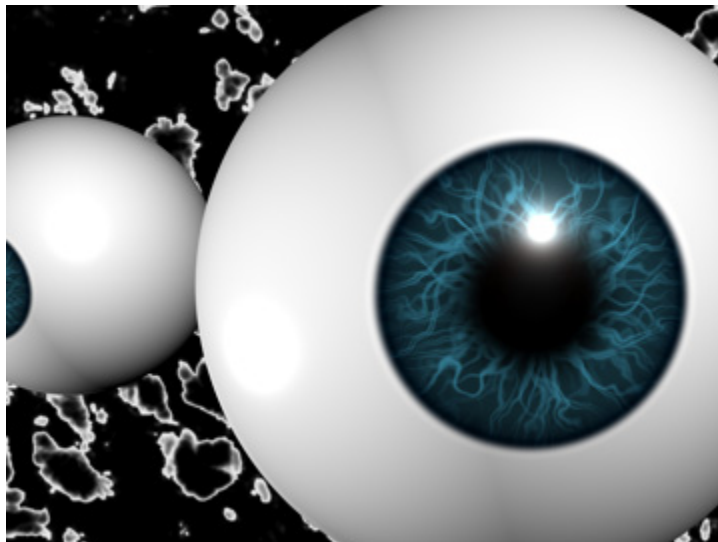
If Television and monitor pixels are different heights, what would a made-for-TV (rectangular) image look like on a square pixel monitor? See for yourself:



(Rendered with TV stretched pixels.)

Notice the roundness of the eye looks a bit squished. Why is that? Because you don't have the tall pixels to display the image as it was intended. Tall pixels would naturally stretch the image to the point that the eye would appear circular.

On a TV, the image would look like this (normal):



(Rendered at square pixels so it looks right on your monitor.)

## DVD Limitations

Although viewing technology continues to improve, many of us are stuck with a standard called DVD which was made for non-square pixels. DVD was designed to support 2 standard aspect ratios, 4x3 (the width and height ratio of the image above, common among older TV's) and 16x9 (widescreen).

The only (NTSC) DVD resolution supported is 720x480 pixels.

As mentioned earlier, both supported aspect ratios use non-square pixels. Newer HDTV's (which have square pixels like a monitor) must stretch and correct the DVD image to compensate.

## The Options

When DVD is phased out, things will no doubt be square across the board, but until then, we have 2 primary choices. (There is no supported square pixel DVD format.)

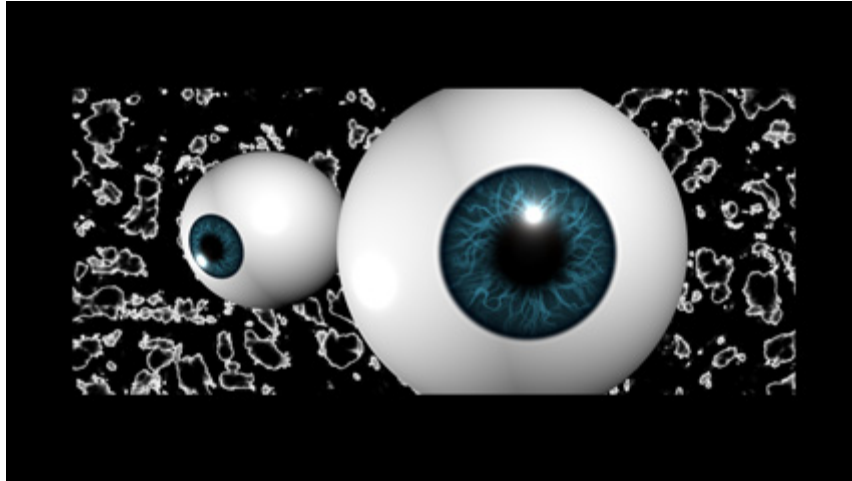
4x3 at 90% squished pixels (.9)  
or  
16x9 at 120% elongated pixels (1.2)

Note: there are other format choices such as interlaced or progressive. Suffice it to say that interlaced is the old standard and progressive is the way of the future. This is beyond the scope of this document. Since I work with film, I choose 24 frames per second progressive (24p) widescreen.

Generally, if you are shooting or rendering at widescreen, or the final product will be widescreen, you will want to choose 16x9 to get the most out of your 720x480 pixels, even if your final product will be wider than 16x9.

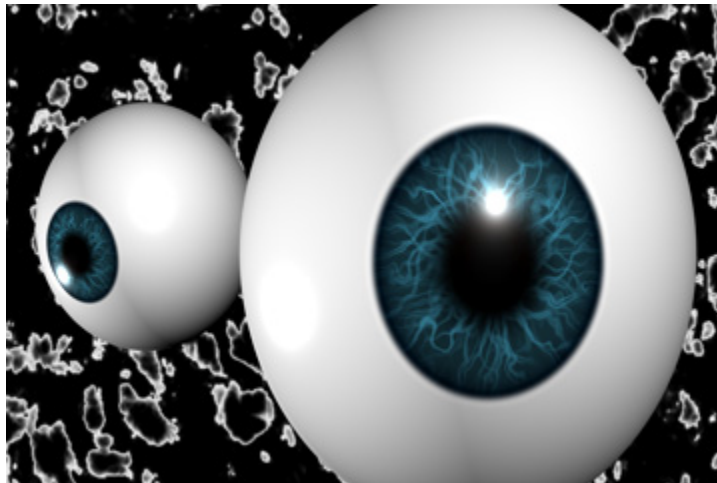
If you know for sure your target audience will be viewing your DVD at 4x3 this may be the solution for you. Alas, both choices can be viewed on either format TV with different side effects.

What about 4x3 letterboxing? Isn't that just as good as widescreen? Absolutely NOT! This may look basically the same on your old 4x3 TV, but on the newer widescreen TV's the 4x3 will appear small and many pixels will be utterly wasted. (See next image.)



(Black bars represent unused 16x9 TV space when using 4x3 letterboxing.)

Below is an example of a 16x9 image. Notice that rather than looking squished down, it is stretched upward 120%.



When played from a DVD player, this image is stretched horizontally (left and right) rather than vertically like the 4x3 format. This means we are still using up all 720x480 pixels. If we were to letterbox in a 4x3 format DVD, we would literally be losing out on 115 pixels just to store black bars along the top and bottom.

Not only does this look beautiful when it fills the entire screen on a wide TV or projector, but when played on a 4x3 TV, it shrinks to appear letterboxed (no unused pixels).

Does this mean we want to render our images at 400 and stretch it 120%? You could, but that would defeat the purpose of using up all 480 pixels. The process of stretching an image in software is kind of like faking those extra pixels and tends to look blurry.

You could render larger, then shrink the image down to exactly the size you want, but then you would be wasting precious rendering time on pixels that would never be seen. Unless you are planning on compiling a full HD 1080p edition later down the road, there is a better way.

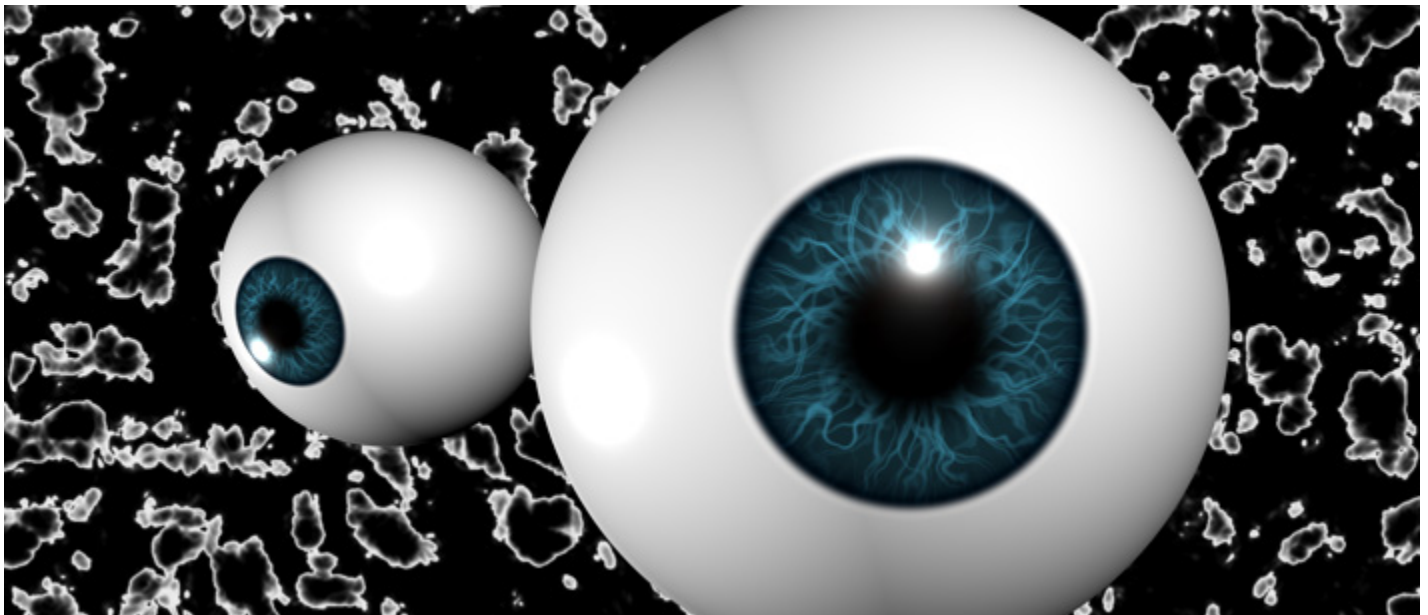
## The Solution

Thanks to the majority of professional rendering software, we can render directly to any stretched format desired, thus providing a perfect transfer to DVD, with no pre-stretching.

For my example I will use a more theatrical aspect ratio of 2.35:1 (wider than 16x9).

We know we want 720 pixels horizontally, but how many pixels vertically would be equivalent to a 2.35:1 aspect ratio?

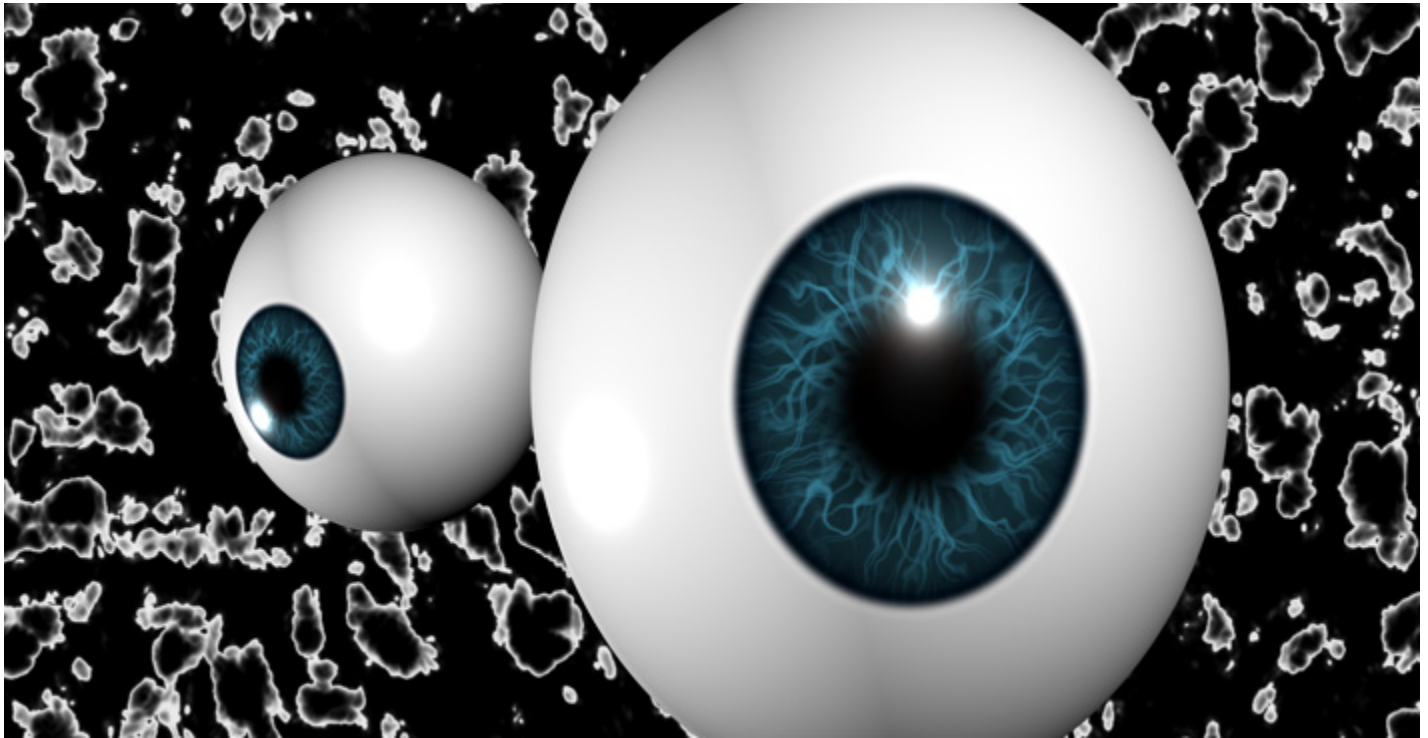
$$720 / 2.35 = 306.3829$$



Rendered at Square Pixels - 720x306 (2.35:1)

However, we must remember that we are still rendering square pixels, and we are not tapping the full potential of our 480 pixel range. If our target media is DVD 16x9, we will want to multiply 306 to the 120% stretch of this format.

$$306.3829 * 1.2 = 367.65948$$



Rendered at - 1.2 Pixels 720x368 (2.35:1)

$480 - 368 = 112$  (unused DVD pixels)

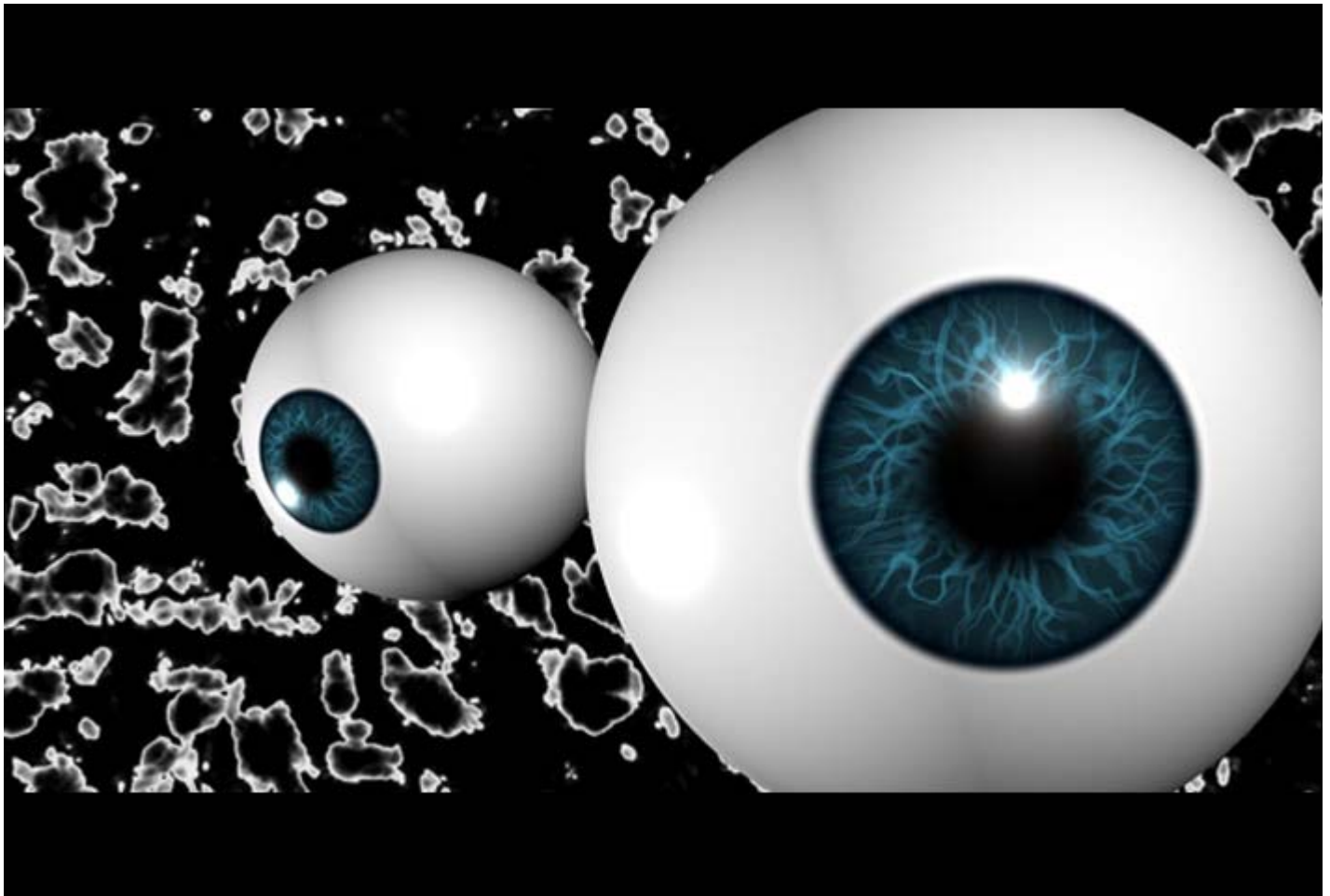
Compared to

$480 - 306 = 174$  (unused DVD pixels)

That's a 62 pixel difference. All that is left is to compare the two images after DVD compression.

## The Comparison

The image below began as the square pixel 720x306 render.

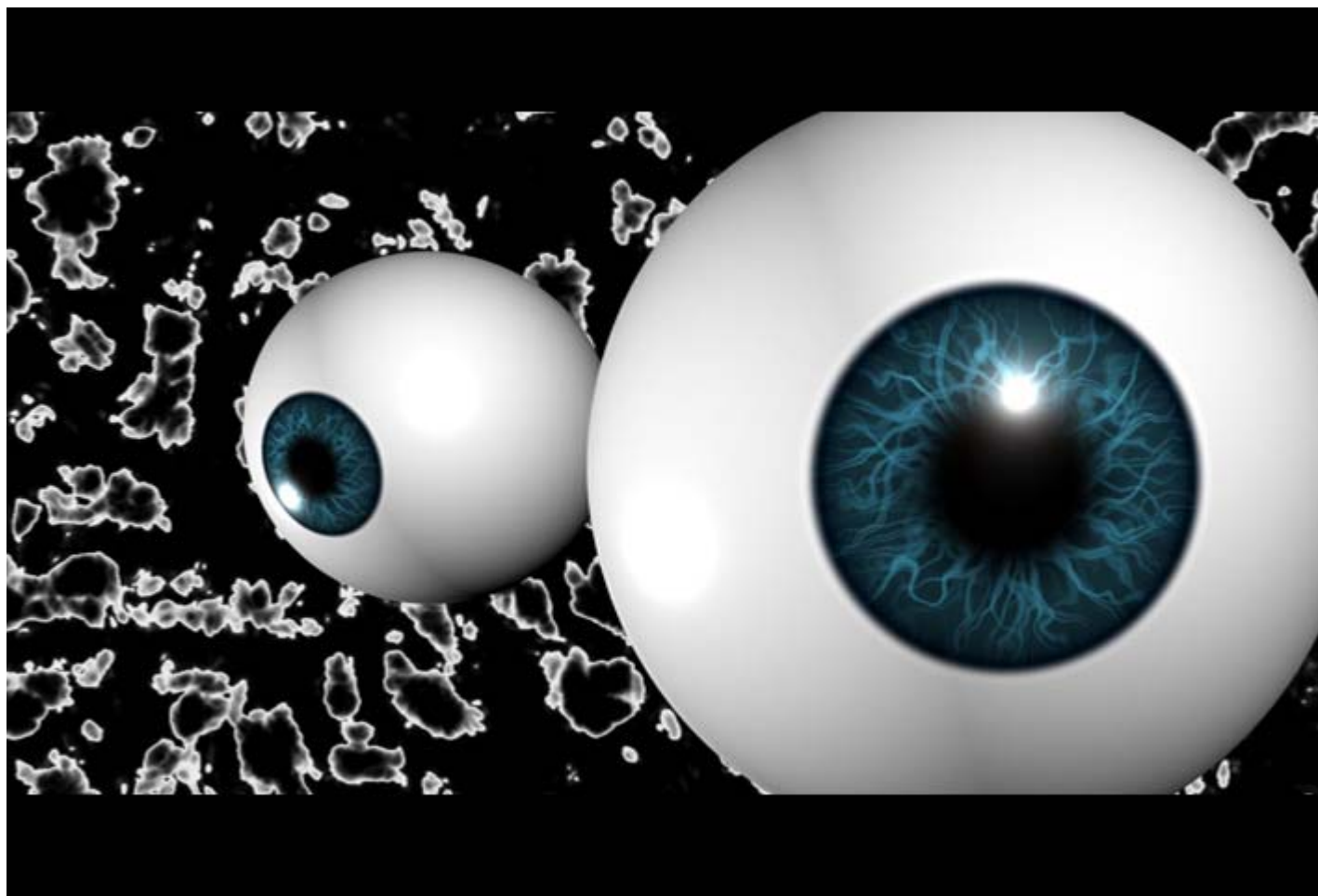


(Image as it would appear on a DVD)

Because this image was originally square pixels, it had to be stretched vertically to match the 1.2 pixel render. In other words, even though we rendered at 720x306 this does not change the fact that, in order for this image to look correct on a DVD, we must stretch the image to 720x368 regardless. Most DVD compression software does this automatically behind the scenes.

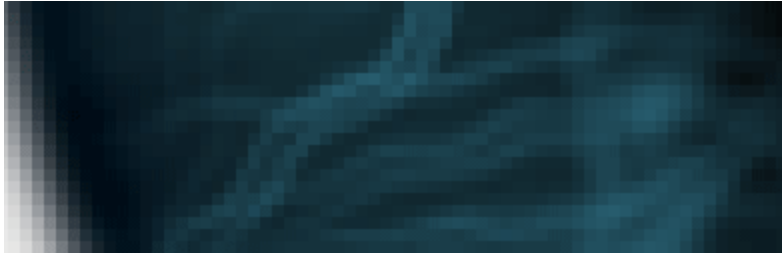


The image below began as the non-square 1.2 pixel 720x368 render.

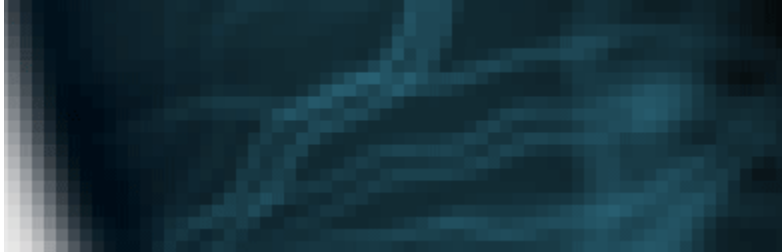


(Note: this image did not need to be stretched to compensate for DVD format.)

At first glance, you might think, wow, not much difference. Let's take a closer look.



(720x306 after DVD stretching.)



(720x368 after DVD stretching.)

The detail of the textures will be hit most as a result of stretching. Portions of the image will appear blurry, and this is before we even get to mpeg compression which will result in even more quality loss. When it comes to compression, it is always best to begin with the highest quality possible, because loss is guaranteed.

Taking advantage of those extra 62 pixels may not seem all that important, but it will add a sense of professional quality as you display your work.

## Final Thoughts

Deciding whether to use this knowledge should be based on the needs of your project.

In practice, I probably would not render using stretched pixels for a large scale production, but for something like a DVD menu. There are some tasks that may be easier to work with when your frames look normal on your monitor, such as editing and compositing.

Also, DVD technology will be phased out very soon and this document may only be of historical value in the short future.

*Please report any comments or errors to [polymesh \(at\) hotmail.com](mailto:polymesh@hotmail.com)*