AI in Colors

From Simple Neural Nets to Large Language Models

DIMITRI REISWICH

8.6.1 Color Embeddings

Imagine a single red light that can shine in many different shades. You can control its brightness with a slider. When the slider is set to zero, the light remains completely dark. As you increase the value, the red glow gradually intensifies until it reaches full brightness. This is illustrated in Figure 8.51, where the slider is positioned at an intensity value of 170. The scale ranges from 0 to 255, allowing us to represent any specific shade of red numerically. A value of 255 represents full brightness, while smaller values correspond to dimmer shades. In this way, a single number precisely defines how bright the red light is.

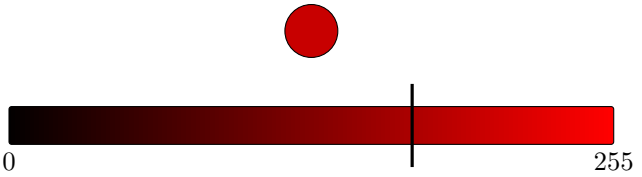

Figure 8.51: Visualization of a red tone with the slider set to an intensity value of 170. The dark-red circle at the top displays the resulting color that corresponds to this setting.

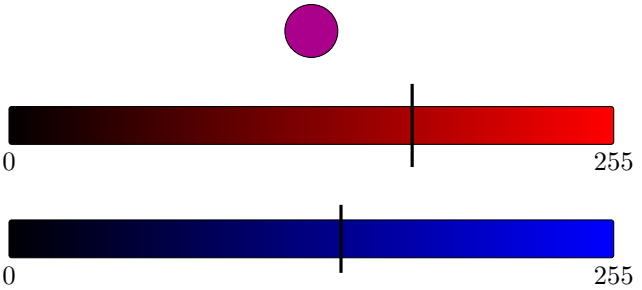

Figure 8.52: Visualization of a color with red intensity 170 and blue intensity 140. The circle displays the corresponding mixed color.

Now let us add a second light, this time blue, with its own brightness setting. Each light can shine independently, but when both are on, their colors mix. By adjusting the brightness of red and blue, we can create an entire range of hues from deep reds to soft purples. Figure 8.52 shows one example: a red intensity of 170 combined with a blue intensity of 140, resulting in a vivid purple represented by the pair $(170, 140)$. This numeric way of representing colors is known as *RGB encoding*. In the full version, we would include a third color, green, to form the complete RGB system. But for now, we will stick with red and blue to keep the concept simple.

Color Similarity Let us now examine the set of colors shown in Figure 8.53. Which of these colors seem most alike? At first glance, it is easy to spot that two of the circles share a shade of red. Likewise, two of the circles display shades of blue that are closely related in tone. Finally, we have purple, which seems to sit somewhere between red and blue. It shares qualities with both but cannot be clearly assigned to either side. Remarkably, you carried out this entire similarity analysis instantly and effortlessly using only your visual perception.

Now let us express this same idea in a different way. Instead of relying on how the colors look, we can represent each color as a point in a geometric space. In such a representation, similar colors are placed close to one another, while dissimilar colors are placed farther apart. The notion of similarity is therefore translated into distance: the closer two points are in this space, the more similar the correspond-

Color	R	B
	255	0
	170	0
	0	255
	0	140
	170	140

Figure 8.53: Colors with their corresponding red/blue values.

ing colors are. In this sense, we replace visual comparison with a distance-based comparison in a geometric space.

To introduce this idea, let us visualize the colors as points in a two-dimensional coordinate system, as shown in Figure 8.54. In this plot, the x-axis represents the red value from Figure 8.53, while the y-axis represents the blue value. What we see is quite intuitive: colors that look similar also appear close together in this space. The two red tones, for example, form a small cluster, as do the two blue shades. In contrast, a bright red and a bright blue are positioned far apart. In this representation, we can also see that the purple point lies closer to the red tones, a relationship that might not be immediately apparent when viewing the colors directly.

To formalize this idea, we treat each point in the plane as a vector. This allows us to describe similarity using mathematical tools rather than visual intuition. These vectors will be called *color embeddings* because each color is positioned (or embedded) within a two-dimensional space. We introduce the vectors x^{r} and x^{dr} to denote the red and darkred embeddings of the points shown in Figure 8.53:

$$x^{\mathrm{r}} = \begin{pmatrix} 255 \\ 0 \end{pmatrix} \qquad x^{\mathrm{dr}} = \begin{pmatrix} 170 \\ 0 \end{pmatrix}$$

Similarly, we have

$$x^{\mathrm{b}} = \begin{pmatrix} 0 \\ 255 \end{pmatrix} \qquad x^{\mathrm{db}} = \begin{pmatrix} 0 \\ 140 \end{pmatrix} \qquad x^{\mathrm{p}} = \begin{pmatrix} 170 \\ 140 \end{pmatrix}$$

One useful property of these color embeddings is that we can perform arithmetic with them. For example, we have

$$x^{\mathrm{dr}} + x^{\mathrm{db}} = x^{\mathrm{p}} \tag{8.11}$$

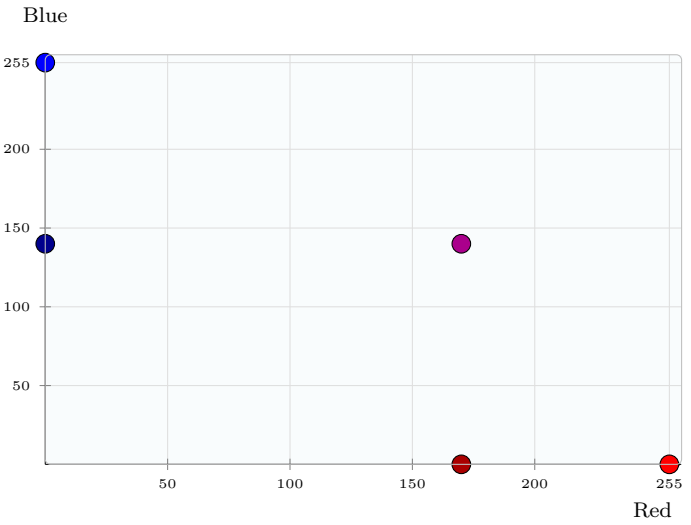

Figure 8.54: Positions of the colors in the (R, B) pane.

This equation has a natural visual interpretation, shown in Figure 8.55. In intuitive

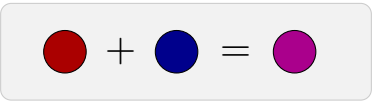

Figure 8.55: Visualization of Equation 8.11.

terms, vector addition corresponds to mixing colored light beams (additive RGB mixing). In the same way, subtracting darkred from purple yields darkblue.

We now turn to numerical distances and metric measures between these vectors, which allow us to quantify how similar or different the colors are from one another.

8.6.2 Euclidean Distance

We begin by defining how to measure the *distance* between embedding vectors. What we are looking for is a function, which we will call d, that takes two input vectors and returns a single number representing how far apart they are in space. A simple and intuitive choice for the vectors x^{r} and x^{dr} is to look at the difference between their first components, since the second component is zero in both cases. This difference is then just the length of the line that connects the darkred and red points in Figure 8.54. Based on this idea, we can define the distance d as the difference between the first vector elements:

$$d(x^{\mathrm{r}}, x^{\mathrm{dr}}) = x_1^{\mathrm{r}} - x_1^{\mathrm{dr}} = 255 - 170 = 85$$

This is a good start, but swapping the inputs would give a negative distance even though the vectors and their spatial distance stay the same. To avoid this problem, we take the absolute of the difference between the first vector entries:

$$d(x^{\mathrm{r}}, x^{\mathrm{dr}}) = |x_1^{\mathrm{r}} - x_1^{\mathrm{dr}}| = |255 - 170| = 85$$

A useful alternative way to write this, which will become important later, is to square the difference and then undo that squaring with a square root:

$$d(x^{\mathrm{r}}, x^{\mathrm{dr}}) = \sqrt{(x_1^{\mathrm{r}} - x_1^{\mathrm{dr}})^2} = |x_1^{\mathrm{r}} - x_1^{\mathrm{dr}}| \tag{8.12}$$

Let us now look at the distance between the purple vector and the red one, i.e. $d(x^{\mathrm{p}}, x^{\mathrm{r}})$. In Figure 8.54, this is the straight line segment connecting the red and purple dots. The question is how to calculate this length. Now, we all know the internet is full of evil and inappropriate content. One of the most offending things I have ever seen online is a meme that says something like: *"Another day has passed and I haven't used the Pythagorean theorem."* Well, not today. Today, defenders of science, Pythagoras will rise from the ashes and give us a hand.

To see how, look at Figure 8.56, where I added a colored triangle between the points shown in Figure 8.54 and labeled the sides. Our distance $d(x^{\mathrm{p}}, x^{\mathrm{r}})$ is now the length of the gray line c. To compute the length, we use the Pythagorean theorem, which tells us that the square of the gray line relates to the two other sides of the triangle as follows:

$$a^2 + b^2 = c^2$$

Conveniently, we already know the length of the green side a: it is simply the difference between the first coordinates of the red and dark red vectors, which is 85. The orange side b is found in exactly the same way, by taking the difference between the second coordinates, giving 140.

So instead of measuring the diagonal directly, we break it into its horizontal and vertical pieces. Using the Pythagorean theorem, the diagonal length becomes:

$$c = \sqrt{85^2 + 140^2} \approx 163.78$$

This geometric idea can be written more compactly using vectors. The red point is the vector x^{r} and the purple point is the vector x^{p}. If we subtract them, we get a new vector that describes exactly how to move from one point to the other:

$$x^{\mathrm{p}} - x^{\mathrm{r}} = \begin{pmatrix} 170 \\ 140 \end{pmatrix} - \begin{pmatrix} 255 \\ 0 \end{pmatrix} = \begin{pmatrix} -85 \\ 140 \end{pmatrix}$$

The two components of this difference vector are exactly the horizontal and vertical distances we just used in the triangle. So the vector is really just a compact way of storing the two side lengths.

To recover the actual distance, we apply the same Pythagorean idea: square the components, add them, and take the square root. To write this more compactly, we

use the notation $\|x\|$ to mean the length (or magnitude) of a vector x. With this notation, the distance between the purple and red vectors is:

$$\|x^{\mathrm{p}} - x^{\mathrm{r}}\| = \sqrt{(x_1^{\mathrm{p}} - x_1^{\mathrm{r}})^2 + (x_2^{\mathrm{p}} - x_2^{\mathrm{r}})^2}$$

So the general vector formula is nothing new, it is just the Pythagorean theorem written in vector form. This connects directly to our earlier one dimensional formula in Equation 8.12, where there was only one component.

The notation for vector length, written as $\|x\|$, can be remembered quite intuitively. The absolute value of a number, $|x|$, represents its magnitude regardless of whether it is positive or negative. By adding another pair of bars, we extend this concept from numbers to vectors, where $\|x\|$ expresses the magnitude or overall length of the vector. And with that, we get a distance measure by defining our distance d as:

$$d(x^{\mathrm{p}}, x^{\mathrm{r}}) = \|x^{\mathrm{p}} - x^{\mathrm{r}}\|$$

It would only be appropriate to call this the Pythagorean distance, but the most common name for it is `Euclidean distance`. The nice thing about this distance measure is that it generalizes to any number of dimensions. If we have two vectors x^a and x^b of dimension n, we can compute their distance as

$$\|x^a - x^b\| = \sqrt{(x_1^a - x_1^b)^2 + (x_2^a - x_2^b)^2 + ... + (x_n^a - x_n^b)^2} \tag{8.13}$$

This is still Pythagoras. He gave us something that works in any dimension. You can imagine placing a flat triangle in 3D space. Each corner of the triangle is now a vector with 3 coordinates, one for x, y, z. But we can still measure the length of the lines connecting those points, and the result is still a single scalar value.

Another useful application of this idea is to measure the length of a single vector rather than the difference of two vectors. If we replace the vector x^b in Equation 8.13 with a vector that contains only zeros (the origin), we get the length of the vector x as

$$\|x\| = \sqrt{x_1^2 + x_2^2 + ... + x_n^2} \tag{8.14}$$

This is called a `vector norm`. We can now compute the distances between all color vectors, as shown in Figure 8.57, and use these values to quantify color similarity. From the figure, we observe that the two shades of red (with a distance of 85) are closer to each other than the two shades of blue (distance 115). Interestingly, the color closest to the purple vector is the dark red one, with a distance of 140, which confirms what our visual impression had already suggested.

A final word on Pythagoras. The Euclidean (or Pythagorean) distance is one of the most widely used concepts in many algorithms and is almost certainly running somewhere on your phone right now. So the real online meme should read: *"Another day has gone by, and I've used Pythagoras without realizing it."*

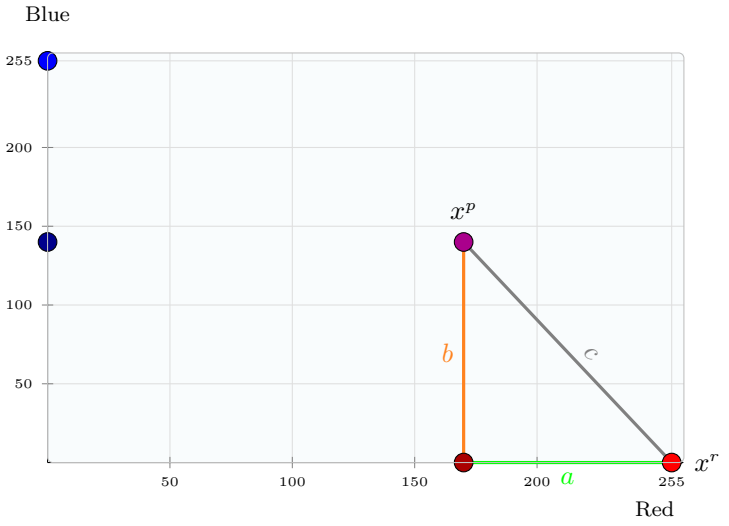

Figure 8.56: Positions of the colors in the (R, B) pane with a triangle used as a basis to calculate the difference between x^{r} and x^{p}.

Pair	Distance
–	85.0
–	360.6
–	290.9
–	163.8
–	306.5
–	220.2
–	140.0
–	115.0
–	205.2
–	170.0

Figure 8.57: Euclidean distances between all pairs of colors.