

MarkToBlue: Illuminating the difference

Alejandro Gombau García

2024-12-23

Introduction.

Immunocytochemistry (IHC) and Immununocytochemistry (ICC) are robust techniques widely used for visualizing antigen locations in tissue samples. While offering unique insights, it is often limited in its quantification capabilities. This project presents a computational solution to complement experimental workflows.

Objectives

The objective of this analysis

1. Explain and document the approach used by MarkToBlue
2. To support the results provided by the program in a detailed way.
3. Explore the possible limitations and propose improvements.

Analitical approach

When using IHC and ICC, we notice that is well-suited for comparative assessment but is of limited value for absolute quantification. Cell samples with different confluence, fluorescence fading, subtle differences in antibody concentration, can make the comparison unreliable.

The proposed method is a ratio creation used for normalize the marker (green) parameters to the blue (base or nuclei) parameters.

The reason for doing so is because we assume that the nuclei marker (DAPI or Hoescht) is a good proxy for total tissue area and brightness baseline.

$$\text{MarkToBlue Ratio} = \frac{A_{\text{blue}} \cdot B_{\text{blue}}}{A_{\text{green}} \cdot B_{\text{green}}}$$

The marker to blue approach has precedents in the literature. Further improvements are needed, however, it may be useful as a first step in marker quantification.

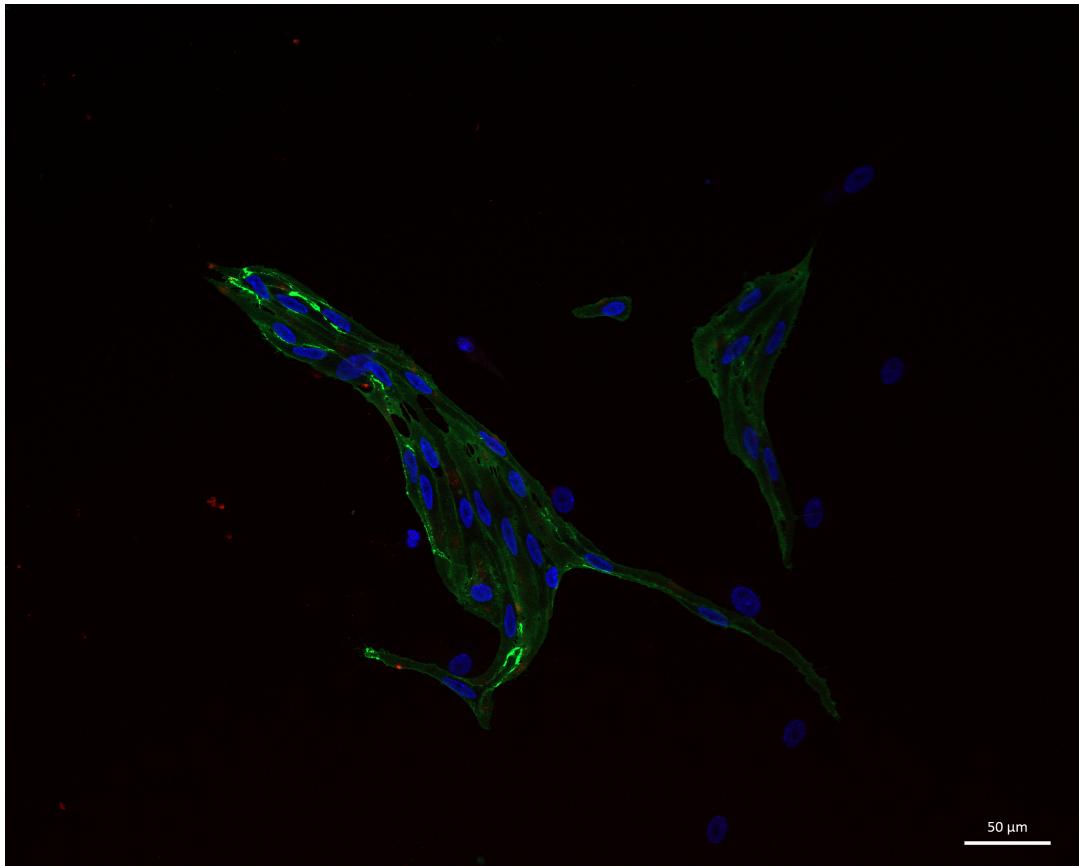
Camp RL, Chung GG, Rimm DL. Automated subcellular localization and quantification of protein expression in tissue microarrays. *Nat Med.* 2002 Nov;8(11):1323-7. doi: 10.1038/nm791. Epub 2002 Oct 21. PMID: 12389040.

Carpenter, A.E., Jones, T.R., Lamprecht, M.R. *et al.* CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biol* 7, R100 (2006). <https://doi.org/10.1186/gb-2006-7-10-r100>

Exploratory Analysis (EA)

We will start the analysis by loading the sample image.

```
path <- "/home/alex/Documentos/Proyectos/marktoblue/MarkToBlue/documentation/sample.png"
# Load the image
image <- readImage(path)
# Display the original image
display(image, method = "raster")
```



After uploading we process the image and plot histograms:

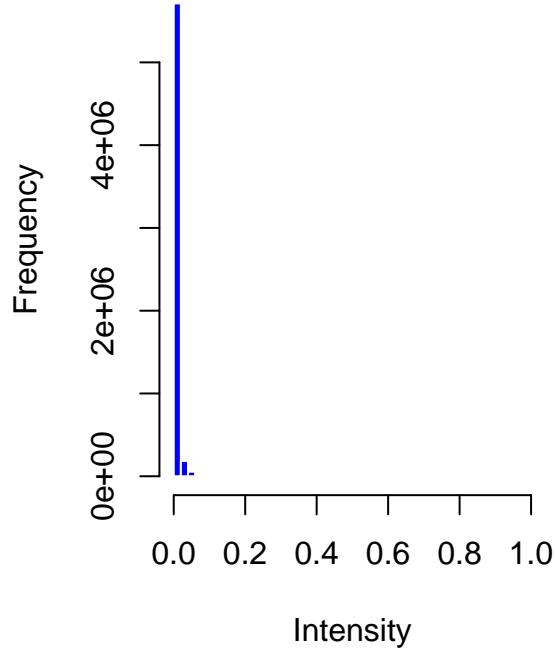
```
# Separate color channels
blue_channel <- channel(image, "blue")
green_channel <- channel(image, "green")

# Plot histograms for color channels
par(mfrow = c(1, 2)) # Arrange plots in a 1x2 grid

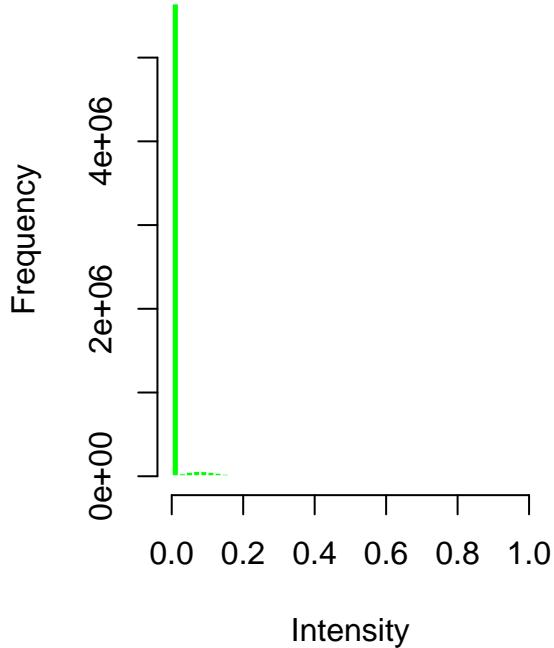
# Histogram for the blue channel
hist(as.vector(blue_channel), breaks = 50, main = "Blue Channel Histogram",
     xlab = "Intensity", col = "blue", border = "white")

# Histogram for the green channel
hist(as.vector(green_channel), breaks = 50, main = "Green Channel Histogram",
     xlab = "Intensity", col = "green", border = "white")
```

Blue Channel Histogram

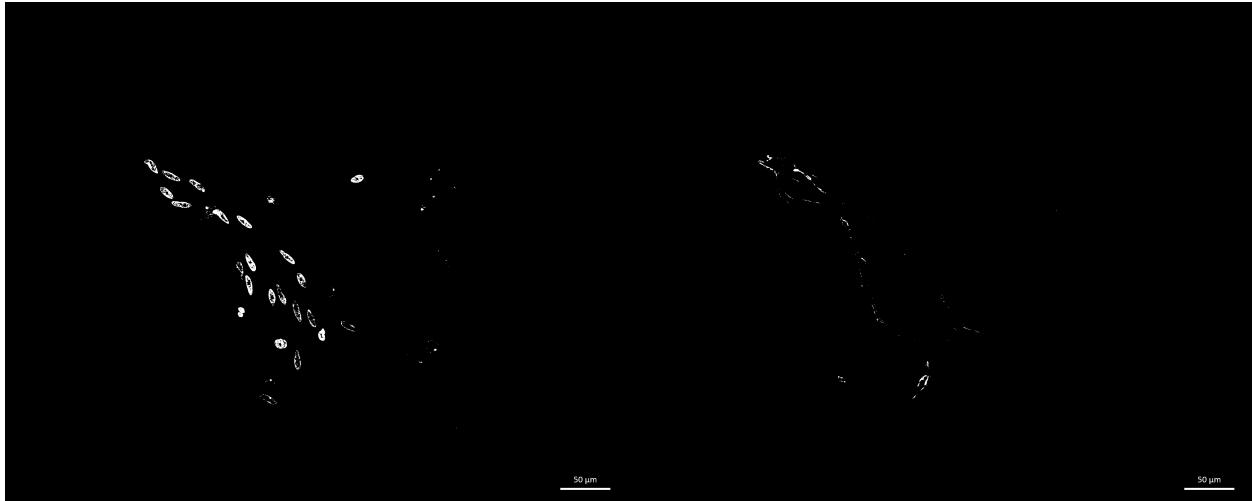


Green Channel Histogram



```
# Thresholding to create masks
threshold_blue <- 0.5 # Adjust as needed
threshold_green <- 0.5 # Adjust as needed
mask_blue <- blue_channel > threshold_blue
mask_green <- green_channel > threshold_green

# Visualize masks
par(mfrow = c(1, 2)) # Arrange plots in a 1x2 grid for masks
display(mask_blue, method = "raster", title = "Blue Mask")
display(mask_green, method = "raster", title = "Green Mask")
```



Calculate area and brightness. But first, the MarkToBlue Shiny App approach is presented.

```
# MarkToBlue App
# Compute metrics
metrics <- reactive({
  req(processed_image())
  channels <- processed_image()

  if (!is.null(channels$green) && !is.null(channels$blue)) {
    # Convert channels to RGB
    green_rgb <- image_convert(channels$green, colorspace = "RGB")
    blue_rgb <- image_convert(channels$blue, colorspace = "RGB")

    # Extract pixel data
    green_matrix <- as.numeric(image_data(green_rgb))
    blue_matrix <- as.numeric(image_data(blue_rgb))

    # Calculate area and brightness
    green_area <- sum(green_matrix > 0)
    blue_area <- sum(blue_matrix > 0)
    # Calculate brightness and multiply by 1000 for legibility
    green_bri <- mean(green_matrix)*1000
    blue_bri <- mean(blue_matrix)*1000

    # Compute MarkToBlue ratio
    marktoblue_ratio <- (green_area * green_bri) / (blue_area * blue_bri)
  }
})
```

Now, we calculate the ratio using the same approach as the Shiny App. We have two variables, `blue_channel` and `green_channel`, each containing pixel values in the range of [0,1]. Each entry correspond to the intensity od the pixel in that channel.

```
# DOCUMENTED R SCRIPT
# Suppose 'blue_channel' and 'green_channel' are numeric vectors or matrices
# that contain pixel values in the [0, 1] range (as returned by image_data())

# 1) Define a "mask" as all non-zero pixels (just like sum(green_matrix > 0))
mask_blue <- (blue_channel > 0)
mask_green <- (green_channel > 0)

# 2) Calculate area (number of non-zero pixels)
area_blue <- sum(mask_blue)
area_green <- sum(mask_green)

# 3) Check for zero area to avoid division by zero
if (area_blue == 0 || area_green == 0) {
  stop("One of the masks has an area of zero. Check your thresholds or input data.")
}

# 4) Calculate mean brightness of only non-zero pixels
brightness_blue <- mean(blue_channel[mask_blue], na.rm = TRUE)
brightness_green <- mean(green_channel[mask_green], na.rm = TRUE)

# 5) Multiply by 1000 for legibility (same approach as R Shiny app)
brightness_blue <- brightness_blue * 1000
brightness_green <- brightness_green * 1000

# 6) Check for zero brightness to avoid invalid results
if (brightness_blue == 0 || brightness_green == 0) {
  stop("One of the channels has a brightness of zero. Check your image processing steps.")
}

# 7) Calculate the Green to Blue ratio (same formula)
gb_ratio <- round((area_green * brightness_green) / (area_blue * brightness_blue), 2)

# 8) Output the ratio
gb_ratio
```

```
## [1] 0.89
```

Below is the result of the same image using MarkToBlue App.



MarkToBlue

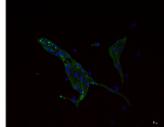
MarkToBlue

Select an image...

 sample.png

Upload complete

Loaded image



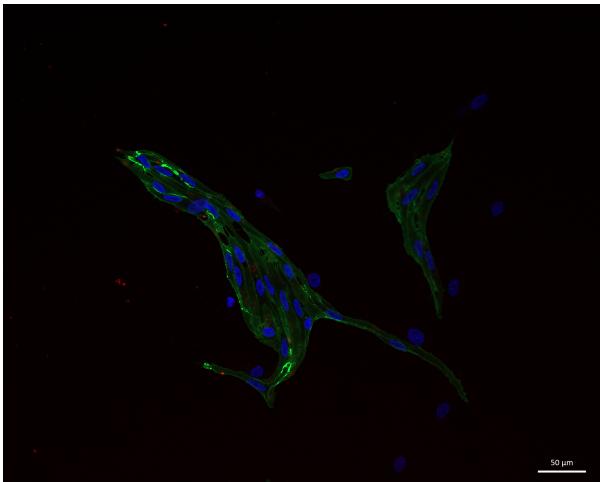
Green area: 1265384
Blue area: 949575
Green Brightness: 2.07
Blue Brightness: 3.02
Mark to Blue RATIO: 0.91

We see that the results differ by ± 0.2 . Probably due to the R rounds results or intermediate variables.

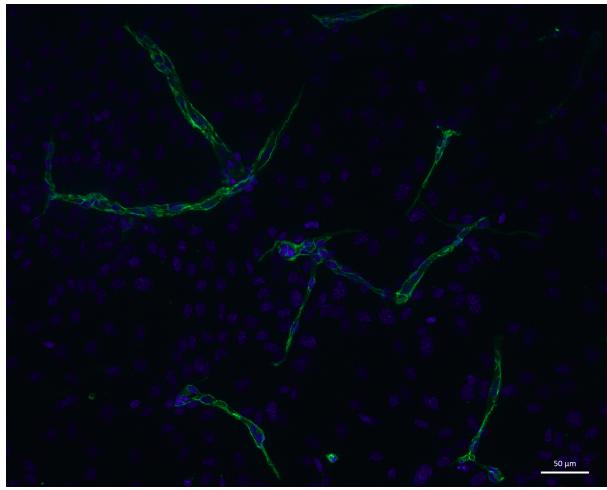
To test whether the app is able to capture experimental differences, two cell culture images are compared. These are brown wistar rat lung cells treated with collagenase and seeded in monolayer. They were then stained with anti Rat CD31/PECAM-1 Antibody(R&D Systems) and Secondary Antibody, Alexa Fluor™ 488(ThermoFisher)

Before seeding, endothelial cells were isolated using the MACS® Columns for magnetic cell isolation (Miltenyi Biotech) system. CD31+ cells were selected.

Isolated cells:



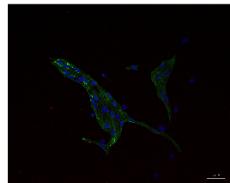
Not isolated cells:



As can be seen, the confluentes are different between the images. You can see how there seems to be less marking in the non-isolated ones. Also the brightness is not the same between the two. This is the expected result when applying antibody based cell isolation techniques. MarkToBlue app will be used to capture experimental differences.

Results for isolated cells

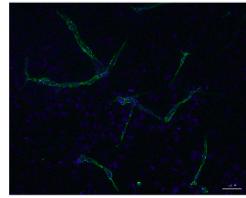
Loaded image



Green area: 1265304
Blue area: 949575
Green Brightness: 2.07
Blue Brightness: 3.02
Mark to Blue RATIO: 0.91

Results for not isolated sample

Loaded image



```
Green area: 9629904
Blue area: 17497914
Green Brightness: 3.91
Blue Brightness: 8.24
Mark to Blue RATIO: 0.26
```

Discussion and future prospects

MarkToBlue has been proven to be able to capture experimental differences. Future statistical analyses need to be performed with more samples to ensure plausibility.

MarkToBlue has an intensive use of libraries can be problematic. There may be transferability problems. That's why we propose the use of dockers or containers for installation and use.