

```
In [25]: import streamlit as st
import numpy as np
from PIL import Image
import matplotlib.pyplot as plt

# Set Streamlit page config
st.set_page_config(page_title="Image Processor", layout="wide")

# Title
st.title("Image - Multi-Color Channel Visualizer")

# Load image from local path
@st.cache_data
def load_image():
    path = r"C:\Users\DELL\Desktop\image.jpg"
    return Image.open(path).convert("RGB")

# Load and display image
elephant = load_image()
st.image(elephant, caption="Original my Image")

# Convert to NumPy array
elephant_np = np.array(elephant)
R, G, B = elephant_np[:, :, 0], elephant_np[:, :, 1], elephant_np[:, :, 2]

# Create channel images
red_img = np.zeros_like(elephant_np)
green_img = np.zeros_like(elephant_np)
blue_img = np.zeros_like(elephant_np)

red_img[:, :, 0] = R
green_img[:, :, 1] = G
blue_img[:, :, 2] = B

# Display RGB channels
st.subheader("RGB Channel Visualization")
col1, col2, col3 = st.columns(3)

with col1:
    st.image(red_img, caption="Red Channel")

with col2:
    st.image(green_img, caption="Green Channel")

with col3:
    st.image(blue_img, caption="Blue Channel")

# Grayscale + Colormap
st.subheader("Colormapped Grayscale Image")

colormap = st.selectbox(
    "Choose a Matplotlib colormap",
    ["viridis", "plasma", "inferno", "magma", "cividis", "hot", "cool", "gray"]
)

elephant_gray = elephant.convert("L")
elephant_gray_np = np.array(elephant_gray)
```

```
# Plot using matplotlib with colormap
fig, ax = plt.subplots(figsize=(6, 4))
ax.imshow(elephant_gray_np, cmap=colormap)
ax.axis("off")
st.pyplot(fig)
```

2025-07-28 15:51:37.018 No runtime found, using MemoryCacheStorageManager

Out[25]: DeltaGenerator()

In []: