

## VANGOGH STYLE

```
import tensorflow_hub as hub
import tensorflow as tf
from PIL import Image
import numpy as np
import matplotlib.pyplot as plt

# Load TF model
hub_model = hub.load("https://tfhub.dev/google/magenta/arbitrary-image-stylization-v1-256/2")

def load_image(path):
    img = Image.open(path).resize((512, 512))
    img = np.array(img) / 255.0
    img = tf.convert_to_tensor(img, dtype=tf.float32)
    return img[tf.newaxis, :]

# Paths (your uploaded files)
content_path = "/content/mountain.jpg"
style_path = "/content/VanGogh_StarryNight01m.jpg"

# Load images
content_image = load_image(content_path)
style_image = load_image(style_path)

# Stylize
stylized_image = hub_model(content_image, style_image)[0]

# Convert for display
stylized_image = tf.squeeze(stylized_image)
stylized_np = np.uint8(stylized_image * 255)

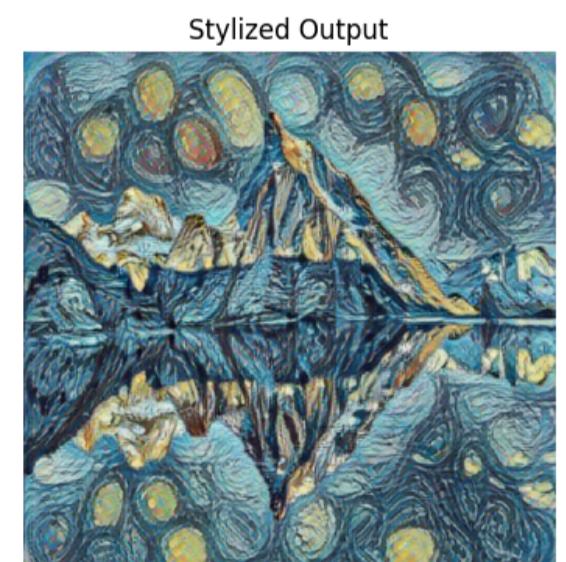
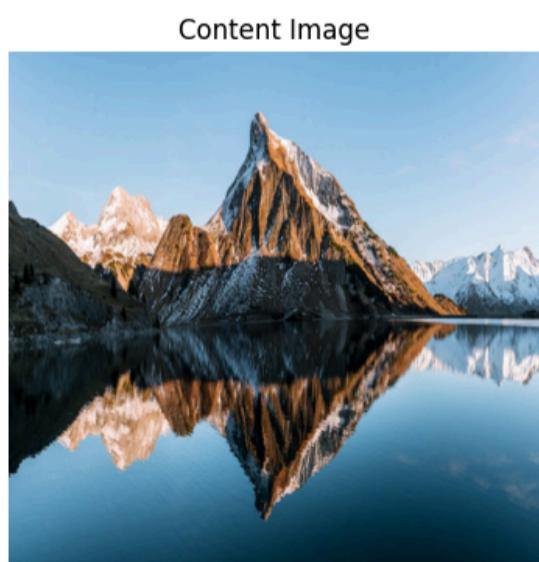
# Show directly in Colab
plt.figure(figsize=(15,5))

plt.subplot(1,3,1)
plt.imshow(np.squeeze(content_image))
plt.title("Content Image")
plt.axis("off")

plt.subplot(1,3,2)
plt.imshow(np.squeeze(style_image))
plt.title("Style Image")
plt.axis("off")

plt.subplot(1,3,3)
plt.imshow(stylized_np)
plt.title("Stylized Output")
plt.axis("off")

plt.show()
```



## MONET STYLE

```
import tensorflow_hub as hub
import tensorflow as tf
from PIL import Image
import numpy as np
```

```

import matplotlib.pyplot as plt

# Load TF model
hub_model = hub.load("https://tfhub.dev/google/magenta/arbitrary-image-stylization-v1-256/2")

def load_image(path):
    img = Image.open(path).resize((512, 512))
    img = np.array(img) / 255.0
    img = tf.convert_to_tensor(img, dtype=tf.float32)
    return img[tf.newaxis, :]

# Paths (your uploaded files)
content_path = "/content/cat.jpeg"
style_path = "/content/monet.jpg"

# Load images
content_image = load_image(content_path)
style_image = load_image(style_path)

# Stylize
stylized_image = hub_model(content_image, style_image)[0]

# Convert for display
stylized_image = tf.squeeze(stylized_image)
stylized_np = np.uint8(stylized_image * 255)

# Show directly in Colab
plt.figure(figsize=(15,5))

plt.subplot(1,3,1)
plt.imshow(np.squeeze(content_image))
plt.title("Content Image")
plt.axis("off")

plt.subplot(1,3,2)
plt.imshow(np.squeeze(style_image))
plt.title("Style Image")
plt.axis("off")

plt.subplot(1,3,3)
plt.imshow(stylized_np)
plt.title("Stylized Output")
plt.axis("off")

plt.show()

```



PICASSO STYLE

```

import tensorflow_hub as hub
import tensorflow as tf
from PIL import Image
import numpy as np
import matplotlib.pyplot as plt

# Load TF model
hub_model = hub.load("https://tfhub.dev/google/magenta/arbitrary-image-stylization-v1-256/2")

def load_image(path):
    img = Image.open(path).resize((512, 512))
    img = np.array(img) / 255.0
    img = tf.convert_to_tensor(img, dtype=tf.float32)

```

```

        return img[tf.newaxis, :]

# Paths (your uploaded files)
content_path = "/content/doll.jpg"
style_path = "/content/picasso.jpg"

# Load images
content_image = load_image(content_path)
style_image = load_image(style_path)

# Stylize
stylized_image = hub_model(content_image, style_image)[0]

# Convert for display
stylized_image = tf.squeeze(stylized_image)
stylized_np = np.uint8(stylized_image * 255)

# Show directly in Colab
plt.figure(figsize=(15,5))

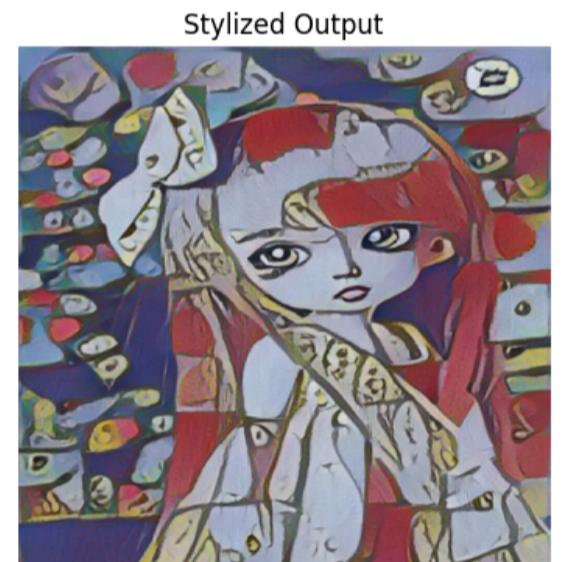
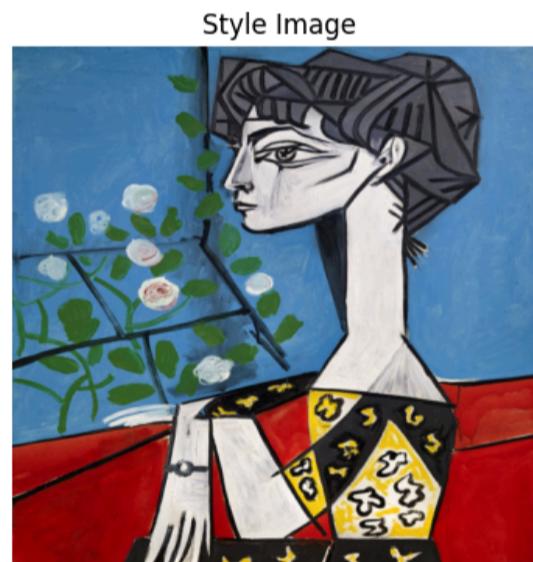
plt.subplot(1,3,1)
plt.imshow(np.squeeze(content_image))
plt.title("Content Image")
plt.axis("off")

plt.subplot(1,3,2)
plt.imshow(np.squeeze(style_image))
plt.title("Style Image")
plt.axis("off")

plt.subplot(1,3,3)
plt.imshow(stylized_np)
plt.title("Stylized Output")
plt.axis("off")

plt.show()

```



## UDNIE STYLE

```

import tensorflow_hub as hub
import tensorflow as tf
from PIL import Image
import numpy as np
import matplotlib.pyplot as plt

# Load TF model
hub_model = hub.load("https://tfhub.dev/google/magenta/arbitrary-image-stylization-v1-256/2")

def load_image(path):
    img = Image.open(path).resize((512, 512))
    img = np.array(img) / 255.0
    img = tf.convert_to_tensor(img, dtype=tf.float32)
    return img[tf.newaxis, :]

# Paths (your uploaded files)
content_path = "/content/cat.jpeg"
style_path = "/content/udnie.jpg"

# Load images
content_image = load_image(content_path)
style_image = load_image(style_path)

```

```

# Stylize
stylized_image = hub_model(content_image, style_image)[0]

# Convert for display
stylized_image = tf.squeeze(stylized_image)
stylized_np = np.uint8(stylized_image * 255)

# Show directly in Colab
plt.figure(figsize=(15,5))

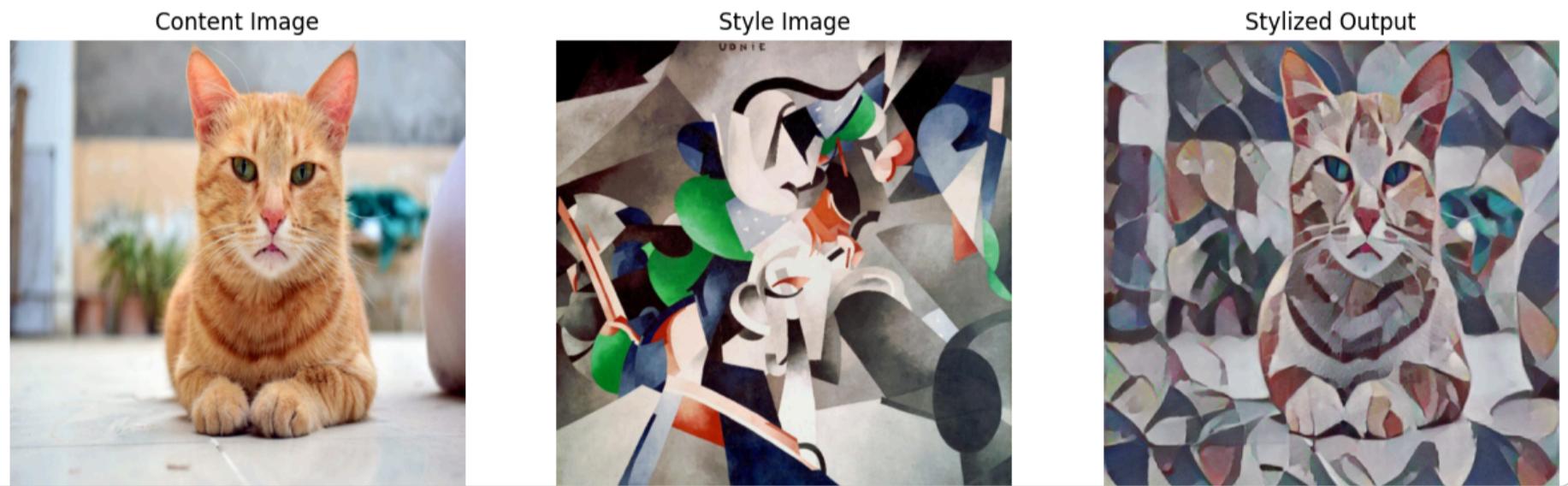
plt.subplot(1,3,1)
plt.imshow(np.squeeze(content_image))
plt.title("Content Image")
plt.axis("off")

plt.subplot(1,3,2)
plt.imshow(np.squeeze(style_image))
plt.title("Style Image")
plt.axis("off")

plt.subplot(1,3,3)
plt.imshow(stylized_np)
plt.title("Stylized Output")
plt.axis("off")

plt.show()

```



## GHIBLI STYLE

```

import tensorflow_hub as hub
import tensorflow as tf
from PIL import Image
import numpy as np
import matplotlib.pyplot as plt

# Load TF model
hub_model = hub.load("https://tfhub.dev/google/magenta/arbitrary-image-stylization-v1-256/2")

def load_image(path):
    img = Image.open(path).resize((512, 512))
    img = np.array(img) / 255.0
    img = tf.convert_to_tensor(img, dtype=tf.float32)
    return img[tf.newaxis, :]

# Paths (your uploaded files)
content_path = "/content/man.jpg"
style_path = "/content/ghibli2.jpg"

# Load images
content_image = load_image(content_path)
style_image = load_image(style_path)

# Stylize
stylized_image = hub_model(content_image, style_image)[0]

# Convert for display
stylized_image = tf.squeeze(stylized_image)
stylized_np = np.uint8(stylized_image * 255)

# Show directly in Colab
plt.figure(figsize=(15,5))

```

```
plt.subplot(1,3,1)
plt.imshow(np.squeeze(content_image))
plt.title("Content Image")
plt.axis("off")

plt.subplot(1,3,2)
plt.imshow(np.squeeze(style_image))
plt.title("Style Image")
plt.axis("off")

plt.subplot(1,3,3)
plt.imshow(stylized_np)
plt.title("Stylized Output")
plt.axis("off")

plt.show()
```

Content Image



Style Image



Stylized Output

