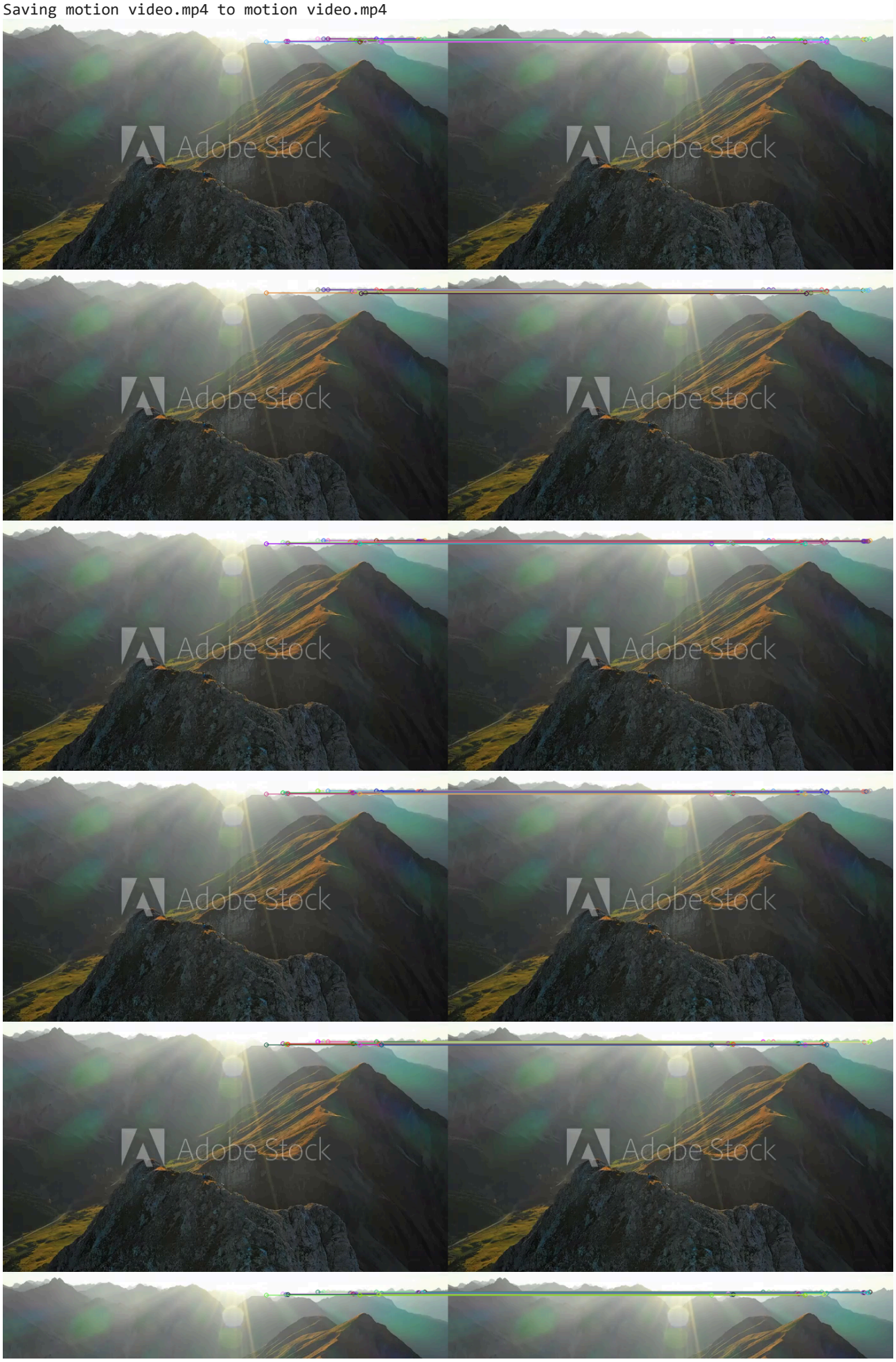
```
# Install OpenCV if needed
!pip install opencv-python-headless
from google.colab.patches import cv2_imshow
# Import required libraries
import cv2
import numpy as np
import matplotlib.pyplot as plt

# Load drone video (you can upload your own or use sample)
from google.colab import files
uploaded = files.upload()  # Upload a video file (e.g., drone.mp4)

video_path = next(iter(uploaded))
cap = cv2.VideoCapture(video_path)

# Initialize ORB and FAST
orb = cv2.ORB_create()
fast = cv2.FastFeatureDetector_create()

# Read first frame
ret, prev_frame = cap.read()
if not ret:
    print("Failed to read video")
    cap.release()

prev_gray = cv2.cvtColor(prev_frame, cv2.COLOR_BGR2GRAY)
prev_kp = fast.detect(prev_gray, None)
prev_kp, prev_des = orb.compute(prev_gray, prev_kp)

while cap.isOpened():
    ret, frame = cap.read()
    if not ret:
        break

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    # Detect keypoints with FAST and compute descriptors with ORB
    kp = fast.detect(gray, None)
    kp, des = orb.compute(gray, kp)

    # Match descriptors using BFMatcher
    if prev_des is not None and des is not None:
        bf = cv2.BFMatcher(cv2.NORM_HAMMING, crossCheck=True)
        matches = bf.match(prev_des, des)
        matches = sorted(matches, key=lambda x: x.distance)

        # Draw top matches
        match_img = cv2.drawMatches(prev_frame, prev_kp, frame, kp, matches[:20], None, flags=2)

        # Display output
        cv2_imshow(match_img)


    prev_frame = frame
    prev_gray = gray
    prev_kp = kp
    prev_des = des

cap.release()
cv2.destroyAllWindows()
```

```
Requirement already satisfied: opencv-python-headless in /usr/local/lib/python3.11/dist-packages (4.
Requirement already satisfied: numpy>=1.21.2 in /usr/local/lib/python3.11/dist-packages (from opencv
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving motion video.mp4 to motion video.mp4
```




Adobe Stock