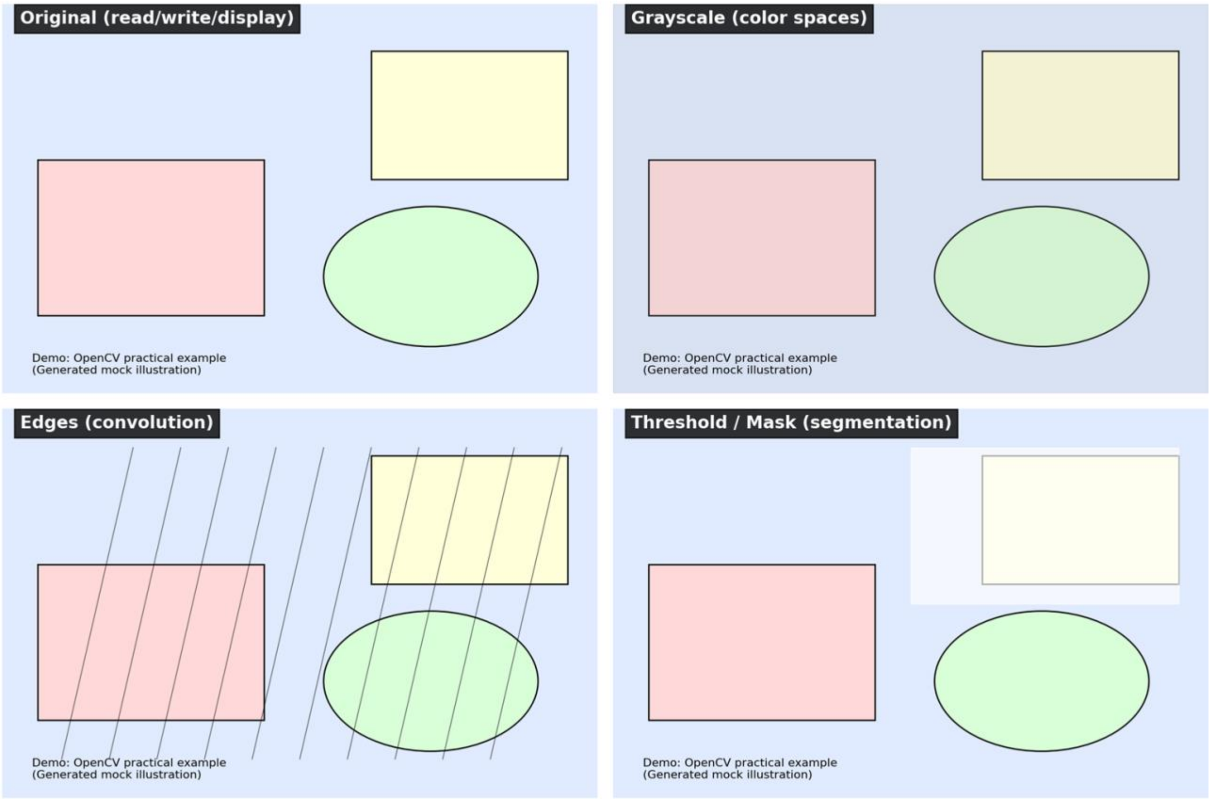
CyArt


Original (read/write/display)
Grayscale (color spaces)
Edges (convolution)
Threshold / Mask (segmentation)
Demo: OpenCV practical example
(Generated mock illustration)
Demo: OpenCV practical example
(Generated mock illustration)
Demo: OpenCV practical example
(Generated mock illustration)
Demo: OpenCV practical example
(Generated mock illustration)

opencv-week1 — Code & Practical Files

README.md

```
# opencv-week1

Practice repository for OpenCV Fundamentals — Week 1.

## Structure

- `notes/` - study notes and links
- `exercises/` - practical scripts demonstrating OpenCV modules
- `research/` - application summaries and platform differences
- `runs/` - outputs and screenshots

## Quickstart
```

1. Create a virtualenv and install requirements:

```bash
python -m venv venv
source venv/bin/activate
pip install -r requirements.txt
```

2. Run example scripts in exercises/:

```
python exercises/01_read_write_display.py
python exercises/02_imgproc_examples.py
python exercises/03_features_orb.py
```

Notes

## requirements.txt

```text
opencv-python
numpy
matplotlib
scikit-image
scikit-learn
```

File: exercises/01_read_write_display.py

```python
"""
Basic I/O and display with OpenCV
- Read an image
- Convert color spaces
- Show and save results
"""
import cv2
import sys
import argparse

def main(image_path: str):
img = cv2.imread(image_path)
if img is None:
print('Failed to read image:', image_path)
return
# Show original
cv2.imshow('Original', img)
# Convert to grayscale
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

```
cv2.imshow('Grayscale', gray)
# Save outputs
cv2.imwrite('runs/original.png', img)
cv2.imwrite('runs/grayscale.png', gray)
print('Saved runs/original.png and runs/grayscale.png')
print('Press any key to exit windows...')
cv2.waitKey(0)
cv2.destroyAllWindows()

if __name__ == '__main__':
parser = argparse.ArgumentParser()
parser.add_argument('image', nargs='?', default='data/sample.jpg', help='Path to an input image')
args = parser.parse_args()
main(args.image)
```

File: exercises/02_imgproc_examples.py

```
"""
Image processing examples:
- Resize, rotate
- Blur, GaussianBlur, MedianBlur
- Edge detection (Sobel, Canny)
- Histogram equalization
"""
import cv2
import numpy as np

img = cv2.imread('data/sample.jpg')
if img is None:
# fallback synthetic image
img = np.zeros((400,600,3), dtype=np.uint8)
cv2.rectangle(img, (50,100),(350,300),(200,100,100), -1)
cv2.putText(img, 'OpenCV', (60,170), cv2.FONT_HERSHEY_SIMPLEX, 2, (255,255,255), 3)

# Resize
resized = cv2.resize(img, (320,240))
# Blur
gauss = cv2.GaussianBlur(resized, (7,7), 1.5)
median = cv2.medianBlur(resized, 5)
# Edges
gray = cv2.cvtColor(resized, cv2.COLOR_BGR2GRAY)
sobelx = cv2.Sobel(gray, cv2.CV_64F, 1, 0, ksize=3)
abs_sobelx = cv2.convertScaleAbs(sobelx)
canny = cv2.Canny(gray, 50, 150)
# Histogram
hist = cv2.calcHist([gray], [0], None, [256], [0,256])
```

```
cv2.imwrite('runs/resized.png', resized)
cv2.imwrite('runs/gaussian.png', gauss)
cv2.imwrite('runs/median.png', median)
cv2.imwrite('runs/sobelx.png', abs_sobelx)
cv2.imwrite('runs/canny.png', canny)

print('Saved processed images to runs/')
```

File: exercises/03_features_orb.py

```
"""
Feature detection & matching using ORB
"""
import cv2
import numpy as np

img1 = cv2.imread('data/sample.jpg', 0)
img2 = cv2.imread('data/sample2.jpg', 0)
if img1 is None or img2 is None:
# create synthetic pair
img1 = np.zeros((300,400), dtype=np.uint8)
cv2.circle(img1, (200,150), 80, 255, -1)
img2 = cv2.flip(img1, 1)

orb = cv2.ORB_create(500)
kp1, des1 = orb.detectAndCompute(img1, None)
kp2, des2 = orb.detectAndCompute(img2, None)

bf = cv2.BFMatcher(cv2.NORM_HAMMING, crossCheck=True)
matches = bf.match(des1, des2)
matches = sorted(matches, key=lambda x: x.distance)

matched = cv2.drawMatches(img1, kp1, img2, kp2, matches[:30], None, flags=2)
cv2.imwrite('runs/orb_matches.png', matched)
print('Saved runs/orb_matches.png')
```

File: exercises/04_objdetect_haar.py

```
"""
Simple object detection using Haar cascade (face detection example)
"""
import cv2

face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_frontalface_default.xm
l')
```

```
img = cv2.imread('data/people.jpg')
if img is None:
print('Provide data/people.jpg with faces to run this demo')
else:
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
faces = face_cascade.detectMultiScale(gray, 1.1, 4)
for (x,y,w,h) in faces:
cv2.rectangle(img, (x,y), (x+w, y+h), (0,255,0), 2)
cv2.imwrite('runs/faces_detected.png', img)
print('Saved runs/faces_detected.png')
```

File: exercises/05_camera_calibration_demo.py

```
"""
Camera calibration demo (synthetic example)
- Generate chessboard points or load sample calibration images
- Compute camera matrix and distortion coefficients
- Show undistorted example
"""
import cv2
import numpy as np

# This demo uses synthetic points to demonstrate API calls
# For real calibration, capture chessboard images and use detectChessboardCorners
pattern_size = (9,6)
objp = np.zeros((pattern_size[0]*pattern_size[1],3), np.float32)
objp[:,:2] = np.mgrid[0:pattern_size[0], 0:pattern_size[1]].T.reshape(-1,2)

# Suppose we have managed to detect corners on N images
objpoints = [objp for _ in range(10)]
imgpoints = [objp[:,:2] + np.random.randn(*objp[:,:2].shape)*0.1 for _ in range(10)]

img_size = (640, 480)
ret, mtx, dist, rvecs, tvecs = cv2.calibrateCamera(objpoints, imgpoints, img_size, None, None)
print('calibration ret:', ret)
print('camera matrix:\n', mtx)
print('dist coeffs:', dist.ravel())

# Undistort sample (synthetic)
sample = np.zeros((img_size[1], img_size[0], 3), dtype=np.uint8)
und = cv2.undistort(sample, mtx, dist, None, mtx)
cv2.imwrite('runs/undistorted.png', und)
print('Saved runs/undistorted.png')
```

File: exercises/06_dnn_inference.py

```
"""
DNN module demo: load a pre-trained MobileNet-SSD from OpenCV DNN (Caffe model)
Note: model files are not included due to size; download separately.
"""
import cv2
import numpy as np

prototxt = 'models/MobileNetSSD_deploy.prototxt'
model = 'models/MobileNetSSD_deploy.caffemodel'
if not (cv2.os.path.exists(prototxt) and cv2.os.path.exists(model)):
print('Place MobileNetSSD model files in models/ or update paths.')
else:
net = cv2.dnn.readNetFromCaffe(prototxt, model)
img = cv2.imread('data/sample.jpg')
h,w = img.shape[:2]
blob = cv2.dnn.blobFromImage(cv2.resize(img, (300,300)), 0.007843, (300,300), 127.5)
net.setInput(blob)
detections = net.forward()
for i in range(detections.shape[2]):
conf = detections[0,0,i,2]
if conf > 0.5:
idx = int(detections[0,0,i,1])
box = detections[0,0,i,3:7] * np.array([w,h,w,h])
(startX, startY, endX, endY) = box.astype('int')
cv2.rectangle(img, (startX, startY), (endX, endY), (0,255,0), 2)
cv2.imwrite('runs/dnn_detections.png', img)
print('Saved runs/dnn_detections.png')
```

File: exercises/07_stitching_demo.py

```
"""
Simple panorama stitching using cv2.Stitcher (works with overlapping images)
"""
import cv2
import glob

images = []
for f in sorted(glob.glob('data/pano_*.jpg')):
images.append(cv2.imread(f))

if len(images) < 2:
print('Provide at least two images named data/pano_*.jpg for stitching demo')
else:
stitcher = cv2.Stitcher_create()
```

```
status, pano = stitcher.stitch(images)
if status == cv2.Stitcher_OK:
cv2.imwrite('runs/panorama.jpg', pano)
print('Saved runs/panorama.jpg')
else:
print('Stitching failed, status=', status)
```

File: exercises/08_video_processing_demo.py

```
"""
Video I/O and simple motion detection (frame differencing)
"""
import cv2
import numpy as np

cap = cv2.VideoCapture(0)  # use webcam for demo; replace with video file path
if not cap.isOpened():
print('Cannot open webcam or video file. Provide a camera or video file.')
else:
ret, prev = cap.read()
if not ret:
print('Failed to read from capture')
else:
prev_gray = cv2.cvtColor(prev, cv2.COLOR_BGR2GRAY)
frame_count = 0
while True:
ret, frame = cap.read()
if not ret:
break
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
diff = cv2.absdiff(prev_gray, gray)
_, th = cv2.threshold(diff, 25, 255, cv2.THRESH_BINARY)
th = cv2.medianBlur(th, 5)
# overlay motion text
motion_pixels = int(np.sum(th) / 255)
if motion_pixels > 500:
cv2.putText(frame, 'Motion Detected', (10,30), cv2.FONT_HERSHEY_SIMPLEX, 1.0,
(0,0,255), 2)
cv2.imshow('Video', frame)
cv2.imshow('Motion Mask', th)
if cv2.waitKey(1) & 0xFF == ord('q'):
break
prev_gray = gray
frame_count += 1
cap.release()
cv2.destroyAllWindows()
```

File: notes/resources.md

```markdown
# Resources for OpenCV Week 1

- Official OpenCV docs: https://docs.opencv.org/4.x/
- YouTube playlist:
https://www.youtube.com/playlist?list=PLjMXczUzEYcHvw5YYSU92WrY8IwhTuq7p
- Example repo reference: https://github.com/pjreddie
```