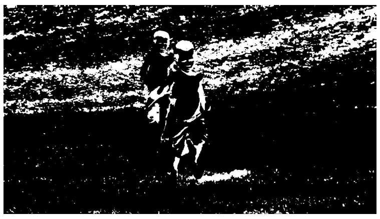
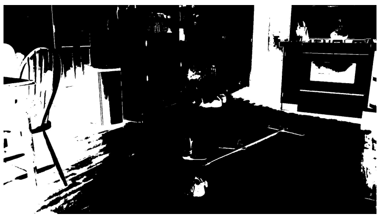
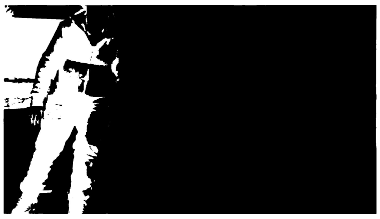
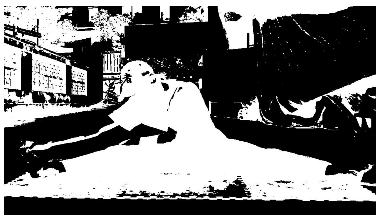
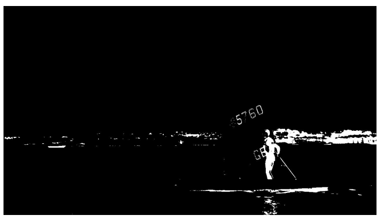
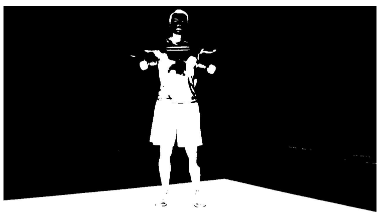
Morphological Operations on MPII Human Pose database

Amir Khayatzadeh

Original Pictures:

Inverted Binary Images with the threshold of 70:

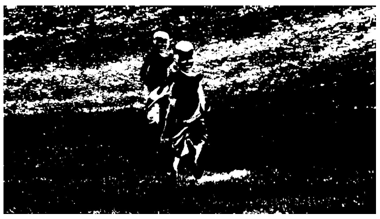
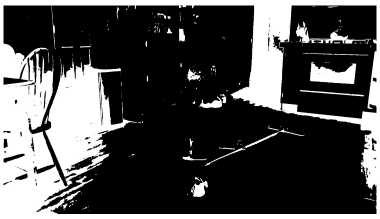
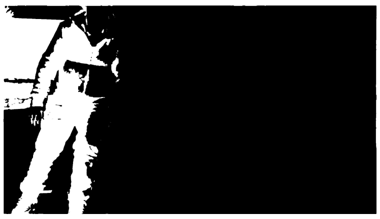
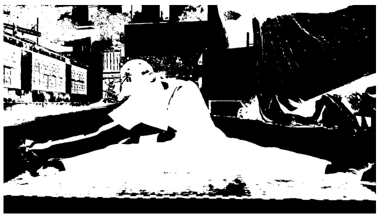
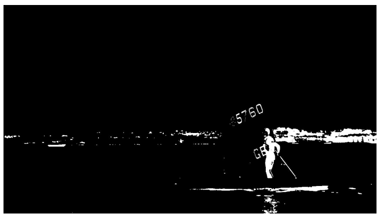
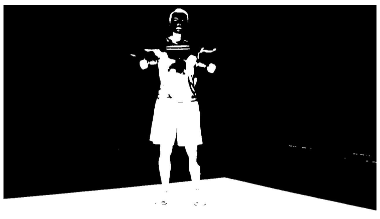
		
		

Produced by using the code:

```
    _, binary_img = cv2.threshold(img, 75, 255, cv2.THRESH_BINARY_INV)
```

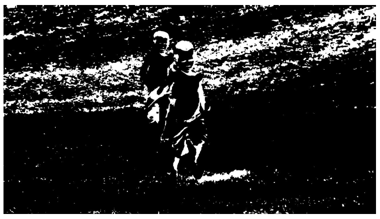
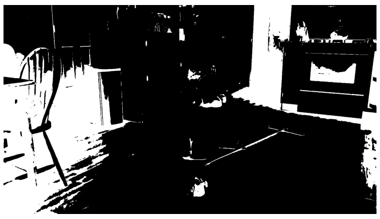
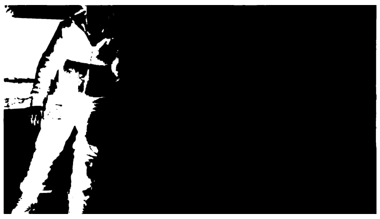
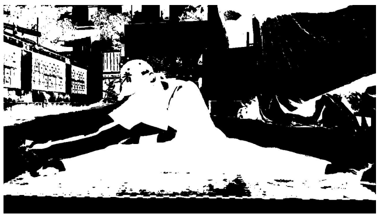
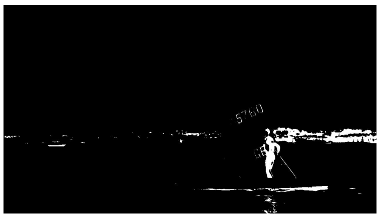
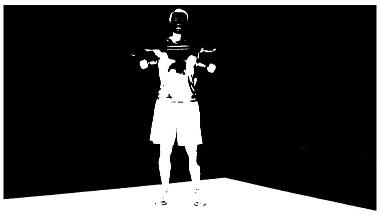
Other thresholds were tested, but none of them worked as intended by making the poses in the images less noticeable. This is still evident in the second picture because the kid's shirt is too bright.

Dilated Images:

These images were produced with these arguments: a (5,5) kernel with 2 iterations.
You can see a gentle difference in the size of white colored pixels; they are wider now. Some deterioration is seen because of this: the feet of the man in the last image are not discernable as they used to be.

Eroded Images:

Arguments: 5 by 5 kernel, and 1 iteration.
The erosion operation has reduced noises in the pictures. Moreover, the feet are now more distinct in the last image. Doing it with one more iteration does not produce different images nor does using 7 X 7 or even 10 X 10 kernels.

The whole source code:

```python
#!/usr/bin/env python3
import cv2
import os

def dilation(binary_img, image_name):
        dilated_img = cv2.dilate(binary_img, (5,5), 2)
        cv2.imwrite(f"{image_name}_dilated.jpg", dilated_img)

def erosion(binary_img, image_name):
        eroded_img = cv2.erode(binary_img, (5,5), 1)
        cv2.imwrite(f"{image_name}_eroded.jpg", eroded_img)


def main():
    images_directory = "./mpii_human_pose/"
    for image_name in os.listdir(images_directory):
        img = cv2.imread(images_directory+image_name,
cv2.IMREAD_GRAYSCALE)

        _, binary_img = cv2.threshold(img, 75, 255,
cv2.THRESH_BINARY_INV)
        cv2.imwrite(f"{image_name}_binary_inv.jpg", binary_img)

        dilation(binary_img, image_name)
        erosion(binary_img, image_name)


if __name__ == "__main__":
    main()
```