```python
import numpy as np
import cv2

# simple threshold
# Adaptive threshold

img1 = cv2.imread('thor.jpg')
img1 = cv2.resize(img1, (600, 600))

img2 = cv2.imread('axe.jpg')
img2 = cv2.resize(img2, (400, 400))

r, c, ch = img2.shape
roi = img1[0:r, 0:c]

# coverting the img2 of into grayscale
gray_img2 = cv2.cvtColor(img2, cv2.COLOR_BGR2GRAY)

#  Now applying the threshold on gray_img2
_, th_gray_img2 = cv2.threshold(gray_img2, 50, 255, cv2.THRESH_BINARY)
# th_gray_img2==mask

# Remove the background
mask_reverse = cv2.bitwise_not(th_gray_img2)

# put the mask into roi
img1_bg = cv2.bitwise_and(roi, roi, mask=mask_reverse)

# take the only region of logo from from the image
img2_fg = cv2.bitwise_and(img2, img2, mask=th_gray_img2)

# put log in ROI and modify the main imag
res = cv2.add(img1_bg, img2_fg)

final = img1
final[0:r, 0:c] = res


# cv2.imshow('img1', img1)
# cv2.imshow('img2', img2)
# cv2.imshow('roi', roi)
# cv2.imshow('gray_img2', gray_img2)
cv2.imshow('th_gray_img2', th_gray_img2)
cv2.imshow('mask_reverse', mask_reverse)
cv2.imshow('img1_bg', img1_bg)
cv2.imshow('img2_fg', img2_fg)
cv2.imshow('res', res)
cv2.imshow('final', final)

cv2.waitKey(0)
```