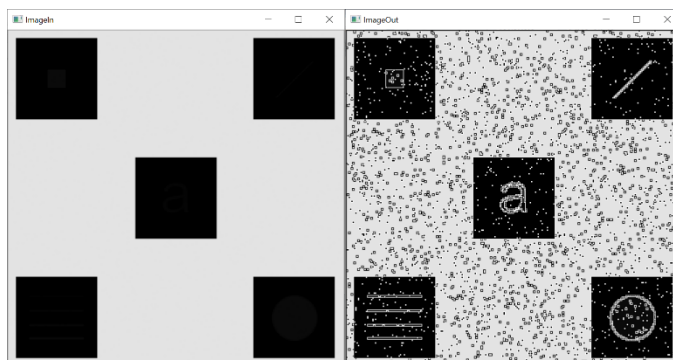


Câu 1: (2đ)

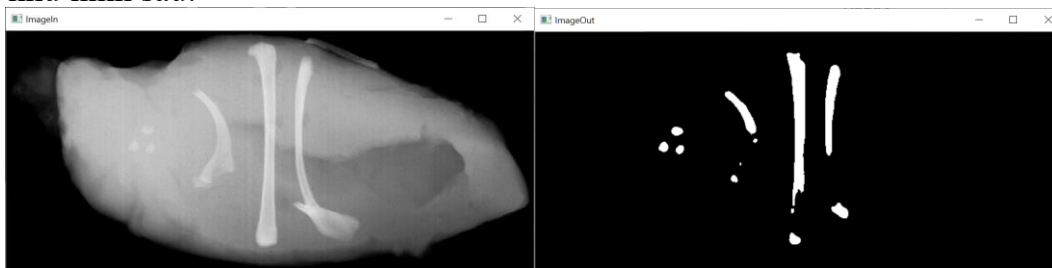
Xây dựng và cài đặt thuật toán làm rõ những đối tượng được nhúng trong năm hình vuông màu đen dùng xử lý histogram cục bộ. (1.5đ)

Tại sao trong trường hợp này cân bằng histogram toàn cục lại không có kết quả? (0.5đ)



Câu 2: (3đ)

Xây dựng và cài đặt thuật toán đếm và tính các điểm ảnh của các mảnh xương trong miếng thịt gà như hình sau:



Câu 3: (3đ)

Implementing the LeNet-5 convolutional neural network architecture to classify MNIST Digits as following:

```
def LeNet_5():
    model = Sequential()
    model.add(Conv2D(filters = 6,
                    kernel_size = 5,
                    strides = 1,
                    activation = 'relu',
                    input_shape = (32,32,1)))
    model.add(MaxPooling2D(pool_size = 2, strides = 2))
    model.add(Conv2D(filters = 16,
                    kernel_size = 5,
                    strides = 1,
                    activation = 'relu',
                    input_shape = (14,14,6)))
    model.add(MaxPooling2D(pool_size = 2, strides = 2))
```

```

model.add(Flatten())
model.add(Dense(units = 120, activation = 'relu'))
model.add(Dense(units = 84, activation = 'relu'))
model.add(Dense(units = 10, activation = 'softmax'))

```

- Draw the network architecture diagram. (1 point)
- Determine the number of parameters of filters (include the bias node) in the convolutional layers. (1 point)
- Determine the number of weights of the fully connected layer. (1 point)

Câu 4: (2đ)

Given the input image and the filter shown in the following figure:

1	2	3	4	5	6	7
11	12	13	14	15	16	17
21	22	23	24	25	26	27
31	32	33	34	35	36	37
41	42	43	44	45	46	47
51	52	53	54	55	56	57
61	62	63	64	65	66	67

1	1	1
1	1	1
1	1	1

- Determine the size of the output image if zero-padding is 0 and stride is 2. (1 point)
- Compute the output image as the convolution of the input image and the filter. (1 point)

-----HẾT-----

Ghi chú: Cán bộ coi thi không giải thích đề thi.

Chuẩn đầu ra của học phần (về kiến thức)	Nội dung kiểm tra
[G 2.1]: <i>Nắm được các phương pháp xử lý ảnh cơ bản</i>	Câu 1, 2, 3, 4
[G 2.2]: <i>Tìm được phương pháp giải quyết một số bài toán xử lý ảnh bằng cách kết hợp một hoặc nhiều phương pháp</i>	Câu 2
[G 3]: <i>Hiểu được các thuật ngữ tiếng Anh cơ bản dùng trong xử lý ảnh</i>	Câu 3, 4

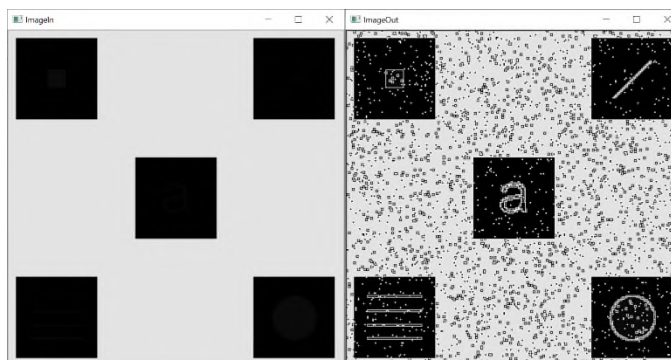
Ngày 15 tháng 01 năm 2021

Thông qua bộ môn
(ký và ghi rõ họ tên)

Câu 1: (2đ)

Xây dựng và cài đặt thuật toán làm rõ những đối tượng được nhúng trong năm hình vuông màu đen dùng xử lý histogram cục bộ. (1.5đ)

Tại sao trong trường hợp này cân bằng histogram toàn cục lại không có kết quả? (0.5đ)

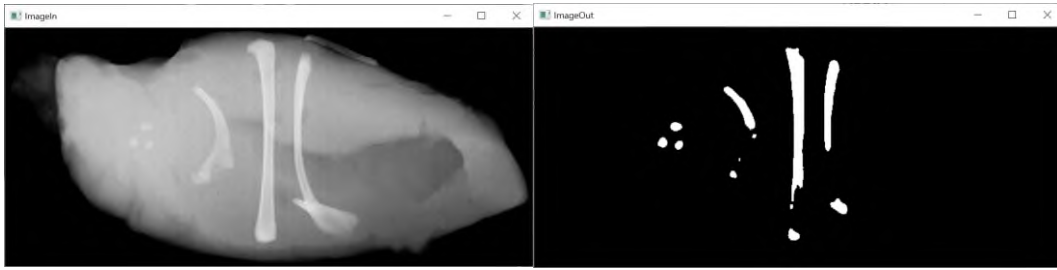


Trả lời:

```
def LocalHist(imgin):  
    M, N = imgin.shape  
    imgout = np.zeros((M,N), np.uint8)  
    m = 3  
    n = 3  
    w = np.zeros((m,n), np.uint8)  
    a = m // 2  
    b = n // 2  
    for x in range(a, M-a):  
        for y in range(b, N-b):  
            for s in range(-a, a+1):  
                for t in range(-b, b+1):  
                    w[s+a,t+b] = imgin[x+s,y+t]  
            w = cv2.equalizeHist(w)  
            imgout[x,y] = w[a,b]  
    return imgout
```

Câu 2: (3đ)

Xây dựng và cài đặt thuật toán đếm và tính các điểm ảnh của các mảnh xương trong miếng thịt gà như hình sau:



Đáp án:

```
def MyConnectedComponent(imgin):
    ret, temp = cv2.threshold(imgin, 200, L-1, cv2.THRESH_BINARY)
    temp = cv2.medianBlur(temp, 7)
    M, N = temp.shape
    dem = 0
    color = 150
    for x in range(0, M):
        for y in range(0, N):
            if temp[x,y] == L-1:
                mask = np.zeros((M+2,N+2),np.uint8)
                cv2.floodFill(temp, mask, (y,x), (color,color,color))
                dem = dem + 1
                color = color + 1
    print('Co %d thanh phan lien thong' % dem)
    a = np.zeros(L, np.int)
    for x in range(0, M):
        for y in range(0, N):
            r = temp[x,y]
            if r > 0:
                a[r] = a[r] + 1
    dem = 1
    for r in range(0, L):
        if a[r] > 0:
            print('%4d %5d' % (dem, a[r]))
            dem = dem + 1
    return temp
```

```
def ConnectedComponent(imgin):
    ret, temp = cv2.threshold(imgin, 200, L-1, cv2.THRESH_BINARY)
    temp = cv2.medianBlur(temp, 7)
    dem, label = cv2.connectedComponents(temp)
    print('Co %d thanh phan lien thong' % (dem-1))

    a = np.zeros(dem, np.int)
    M, N = label.shape
    color = 150
    for x in range(0, M):
        for y in range(0, N):
```

```

r = label[x, y]
a[r] = a[r] + 1
if r > 0:
    label[x,y] = label[x,y] + color

for r in range(1, dem):
    print('%4d %10d' % (r, a[r]))
return label.astype(np.uint8)

```

Câu 3: (3đ)

Implementing the LeNet-5 convolutional neural network architecture to classify MNIST Digits as following:

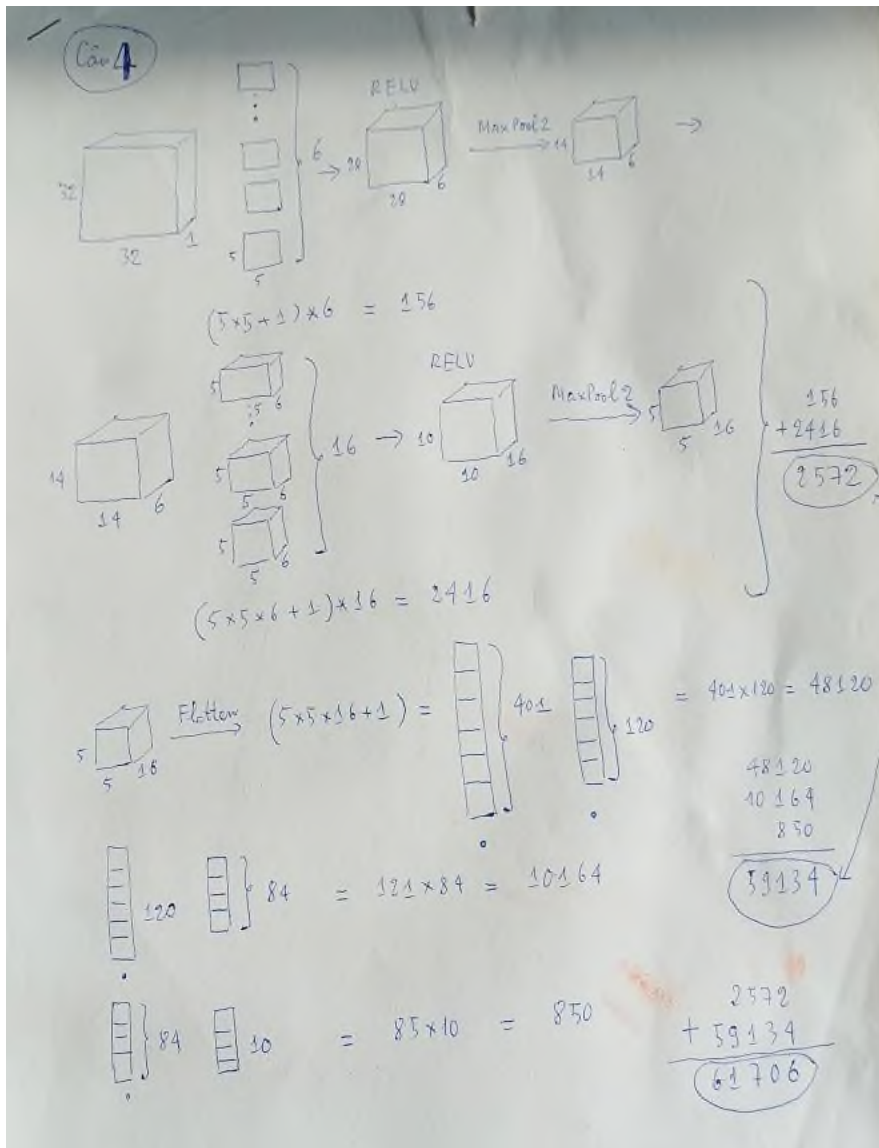
```

def LeNet_5():
    model = Sequential()
    model.add(Conv2D(filters = 6,
                    kernel_size = 5,
                    strides = 1,
                    activation = 'relu',
                    input_shape = (32,32,1)))
    model.add(MaxPooling2D(pool_size = 2, strides = 2))
    model.add(Conv2D(filters = 16,
                    kernel_size = 5,
                    strides = 1,
                    activation = 'relu',
                    input_shape = (14,14,6)))
    model.add(MaxPooling2D(pool_size = 2, strides = 2))
    model.add(Flatten())
    model.add(Dense(units = 120, activation = 'relu'))
    model.add(Dense(units = 84, activation = 'relu'))
    model.add(Dense(units = 10, activation = 'softmax'))

```

- Draw the network architecture diagram. (1 point)
- Determine the number of parameters of filters (include the bias node) in the convolutional layers. (1 point)
- Determine the number of weights of the fully connected layer. (1 point)

Đáp án:



Using TensorFlow backend.

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 28, 28, 6)	156
max_pooling2d_1 (MaxPooling2 (None, 14, 14, 6))		0
conv2d_2 (Conv2D)	(None, 10, 10, 16)	2416
max_pooling2d_2 (MaxPooling2 (None, 5, 5, 16))		0
flatten_1 (Flatten)	(None, 40)	0
dense_1 (Dense)	(None, 120)	48120
dense_2 (Dense)	(None, 84)	10164

dense_3 (Dense) (None, 10) 850

Total params: 61,706

Trainable params: 61,706

Non-trainable params: 0

Câu 4: (2đ)

Given the input image and the filter shown in the following figure:

1	2	3	4	5	6	7
11	12	13	14	15	16	17
21	22	23	24	25	26	27
31	32	33	34	35	36	37
41	42	43	44	45	46	47
51	52	53	54	55	56	57
61	62	63	64	65	66	67

1	1	1
1	1	1
1	1	1

- Determine the size of the output image if zero-padding is 0 and stride is 2. (1 point)
- Compute the output image as the convolution of the input image and the filter. (1 point)

Trả lời:

- $W_{out} = (W-F+2P)/S + 1 = (7-3+2*0)/2 + 1 = 3$, therefore the size of output image is 3x3
-

108	126	144
288	306	324
468	486	504

-----HẾT-----

Ghi chú: Cán bộ coi thi không giải thích đề thi.

Ngày 15 tháng 01 năm 2021

Thông qua bộ môn
(ký và ghi rõ họ tên)