



Descritores de Textura (LBP + GLCM) com classificador k-NN e SVM

Caio Miglioli
Geazz Zanoni

Banco de Imagens

- Base B: BFL 256x256*;
- 03 amostras de 315 autores ;
- Cada amostra foi dividida em 9 partes
- 315 classes, com 27 amostras

*C.Freitas, L.S.Oliveira, R.Sabourin, and F.Bortolozzi, "Brazilian forensic letter database" in 11th International workshop on frontiers on handwriting recognition, Montreal, Canada, 2008

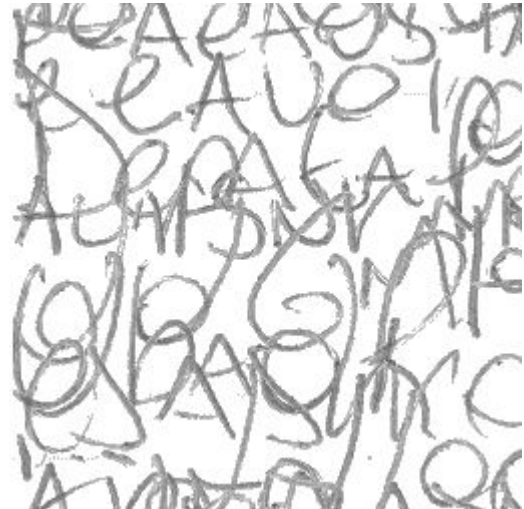


Imagem CF00001_1.bmp

Implementação

Preparação do Banco de Imagens:

- `get_files()` e `createXy()`
 - `imread()`
- `calculate_glcm()`
 - `graycomatrix()` e `graycoprops()`
 - contrast, dissimilarity, homogeneity, ASM, energy e correlation
- `calculate_lbp()`
 - `local_binary_pattern()`
 - 8 neighbors, 2 radius, nri_uniform
- `split_manual()`
 - Alternativa para o `train_test_split()`

```
def createXy(database, quadrants=(1,1), lbp_props={'neighbours': 8, 'radius': 2}):
    X_glcm, X_lbp, y = list(), list(), list()

    for classe in database:
        for item in natsort.natsorted(database[classe]):
            image = cv2.imread(str(item), cv2.IMREAD_GRAYSCALE)

            X_glcm.append(calculate_glcm(image, quadrants))
            X_lbp.append(calculate_lbp(image, quadrants, lbp_props['neighbours'], lbp_props['radius']))
            y.append(classe)

    return np.array(X_glcm), np.array(X_lbp), np.array(y)
#end createxy
```

Função createXy()

```
# Retorna o quadrante da imagem
quadrant = image[left:right, upper:lower]

# Calculate Features
lbp_image = local_binary_pattern(quadrant, neighbours, radius, method="nri_uniform")
n_bins = int(lbp_image.max() + 1)
hist, _ = np.histogram(lbp_image, density=True, bins=n_bins, range=(0, n_bins))

# Append to result
result.extend(hist)

return result
#end lbp
```

Parte final da função calculate_lbp()



Testes e Resultados

Parâmetros:

Quadrantes:

- 1x1, 2x2 e 4x4

k-NN:

- k: 1, 3 e 5
- distância: cityblock e euclidean

SVM:

- kernel: rbf e linear
- C: 0.03125, 2.0 e 128.0
- gamma: 8.0, 0.125 e 0.0019

20 Melhores Resultados

Acurácia	Quadrantes	Descritor	Classificador	Parâmetros
1.0	1	LBP	KNN	quadr=1x1, distance=euclidean, k=3, LBP, KNN
1.0	1	LBP	SVM	quadr=1x1, kernel=rbf, C=128.0, gamma=0.001953125, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=rbf, C=8192.0, gamma=0.001953125, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=0.03125, gamma=8.0, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=0.03125, gamma=0.125, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=0.03125, gamma=0.001953125, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=2.0, gamma=8.0, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=2.0, gamma=0.125, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=2.0, gamma=0.001953125, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=128.0, gamma=8.0, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=128.0, gamma=0.125, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=128.0, gamma=0.001953125, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=8192.0, gamma=8.0, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=8192.0, gamma=0.125, LBP, SVM
1.0	1	LBP	SVM	quadr=1x1, kernel=linear, C=8192.0, gamma=0.001953125, LBP, SVM
0.9875	1	LBP	KNN	quadr=1x1, distance=cityblock, k=1, LBP, KNN
0.9875	1	LBP	KNN	quadr=1x1, distance=euclidean, k=5, LBP, KNN
0.9875	2	LBP	SVM	quadr=2x2, kernel=rbf, C=128.0, gamma=0.001953125, LBP, SVM
0.9875	2	LBP	SVM	quadr=2x2, kernel=rbf, C=8192.0, gamma=0.001953125, LBP, SVM
0.9875	2	LBP	SVM	quadr=2x2, kernel=linear, C=0.03125, gamma=8.0, LBP, SVM

20 Piores Resultados

Acurácia	Quadrantes	Descritor	Classificador	Parâmetros
0.1	4	LBP	SVM	quadr=4x4, kernel=rbf, C=0.03125, gamma=8.0, LBP, SVM
0.1	4	LBP	SVM	quadr=4x4, kernel=rbf, C=2.0, gamma=8.0, LBP, SVM
0.1	4	LBP	SVM	quadr=4x4, kernel=rbf, C=128.0, gamma=8.0, LBP, SVM
0.1	4	LBP	SVM	quadr=4x4, kernel=rbf, C=8192.0, gamma=8.0, LBP, SVM
0.1375	2	LBP	SVM	quadr=2x2, kernel=rbf, C=0.03125, gamma=0.125, LBP, SVM
0.15	2	GLCM	SVM	quadr=2x2, kernel=rbf, C=0.03125, gamma=8.0, GLCM, SVM
0.1625	2	GLCM	SVM	quadr=2x2, kernel=rbf, C=2.0, gamma=8.0, GLCM, SVM
0.1625	2	GLCM	SVM	quadr=2x2, kernel=rbf, C=128.0, gamma=8.0, GLCM, SVM
0.1625	2	GLCM	SVM	quadr=2x2, kernel=rbf, C=8192.0, gamma=8.0, GLCM, SVM
0.225	1	LBP	SVM	quadr=1x1, kernel=rbf, C=2.0, gamma=8.0, LBP, SVM
0.225	1	LBP	SVM	quadr=1x1, kernel=rbf, C=128.0, gamma=8.0, LBP, SVM
0.225	1	LBP	SVM	quadr=1x1, kernel=rbf, C=8192.0, gamma=8.0, LBP, SVM
0.2625	4	GLCM	SVM	quadr=4x4, kernel=rbf, C=0.03125, gamma=0.125, GLCM, SVM
0.275	2	LBP	SVM	quadr=2x2, kernel=rbf, C=0.03125, gamma=8.0, LBP, SVM
0.275	2	LBP	SVM	quadr=2x2, kernel=rbf, C=2.0, gamma=8.0, LBP, SVM
0.275	2	LBP	SVM	quadr=2x2, kernel=rbf, C=128.0, gamma=8.0, LBP, SVM
0.275	2	LBP	SVM	quadr=2x2, kernel=rbf, C=8192.0, gamma=8.0, LBP, SVM
0.375	2	LBP	SVM	quadr=2x2, kernel=rbf, C=2.0, gamma=0.125, LBP, SVM
0.375	2	LBP	SVM	quadr=2x2, kernel=rbf, C=128.0, gamma=0.125, LBP, SVM
0.375	2	LBP	SVM	quadr=2x2, kernel=rbf, C=8192.0, gamma=0.125, LBP, SVM

Muito obrigado!

Dúvidas?