

## Laporan Hasil Analisa Perbandingan Antara Color, Texture, dan Shape Descriptor

Pada eksperimen ini saya menggunakan LBP untuk *texture descriptor*. Kemudian untuk *color descriptor* saya melakukan penghitungan nilai rata-rata suatu pixel untuk tiga dimensi yaitu RGB. Dan untuk *shape/edge descriptor* saya menggunakan *prewitt transform*. Dari tiga metode tersebut, saya melakukan analisa pada tiga citra berikut :

1



2

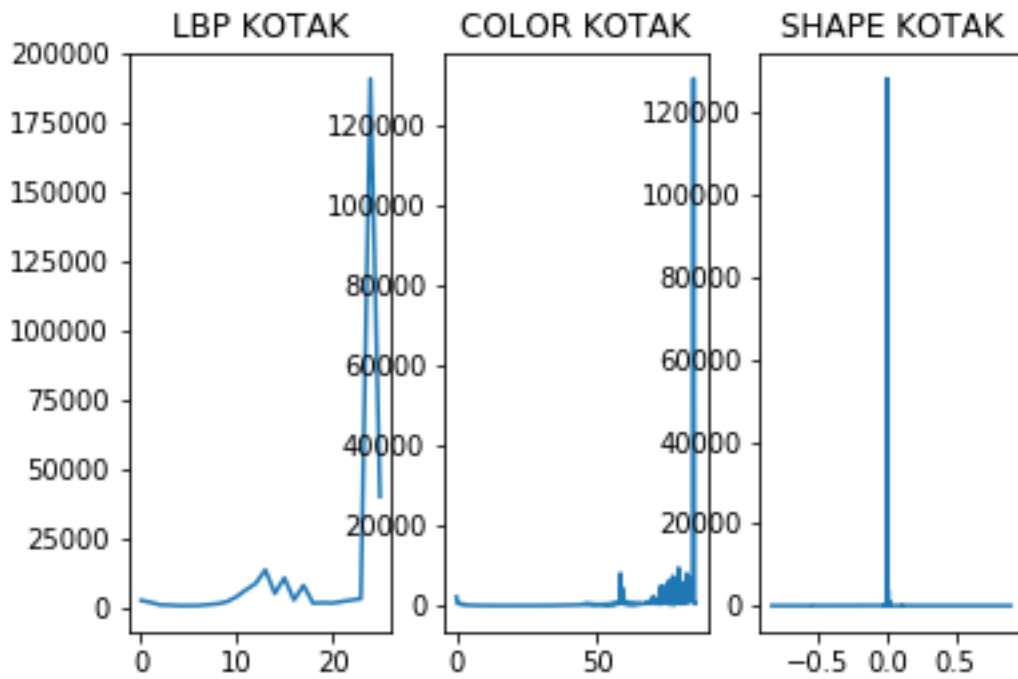


3

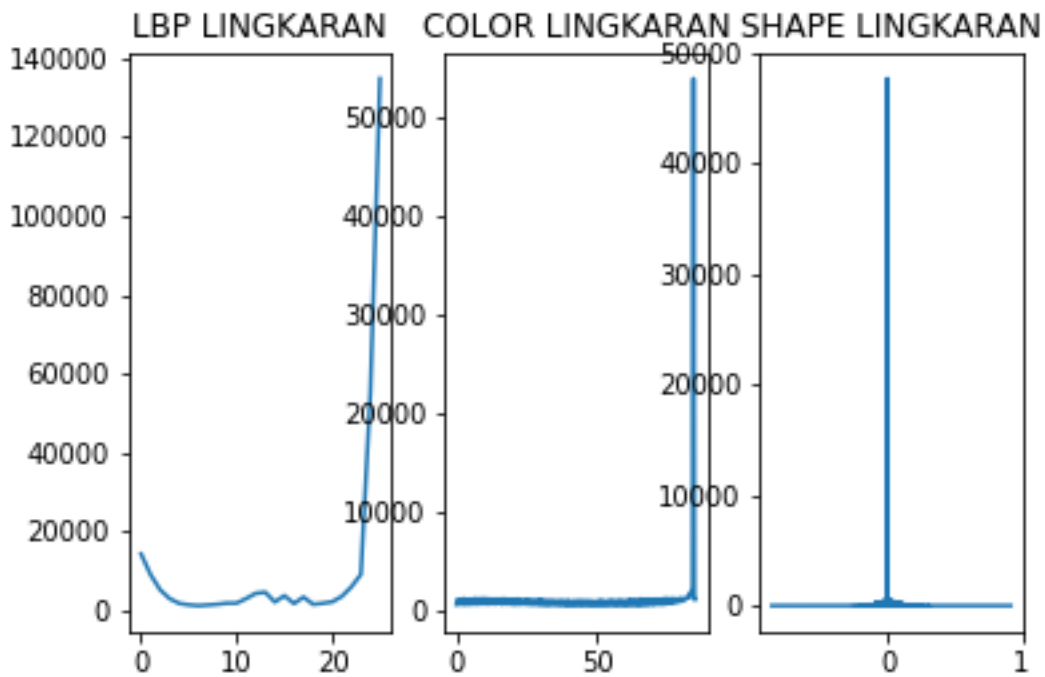


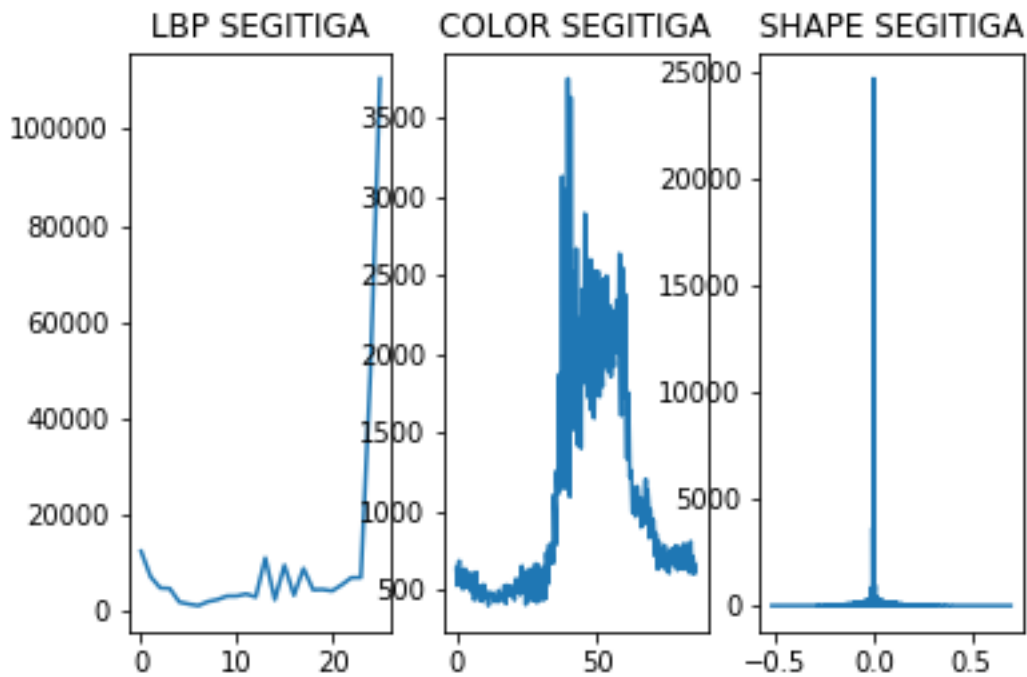
Dari tiga citra tersebut saya mendapatkan histogram fitur sebagai berikut :

1



2





Dan berikut adalah rangkuman nilai *Euclidean Distance* tiap fitur hasil esktraksi :

```
print("Berikut nilai euclidean untuk setiap feature extraction citra 1 (Buku | Kotak) : ")
print("LBP :",euclideanDistance(lbpKotak))
print("COLOR :",euclideanDistance(colorKotak))
print("SHAPE :",euclideanDistance(shapeKotak))
```

```
Berikut nilai euclidean untuk setiap feature extraction citra 1 (Buku | Kotak) :
LBP : 12283.74035056098
COLOR : 42479.877678209
SHAPE : 41.999018803036186
```

```
print("Berikut nilai euclidean untuk setiap feature extraction citra 2 (Koin | Lingkaran) : ")
print("LBP :",euclideanDistance(lbpLingkaran))
print("COLOR :",euclideanDistance(colorLingkaran))
print("SHAPE :",euclideanDistance(shapeLingkaran))
```

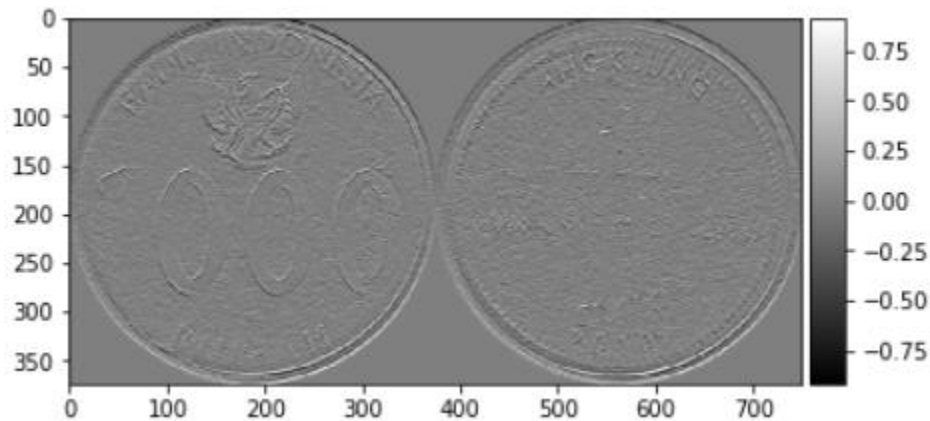
```
Berikut nilai euclidean untuk setiap feature extraction citra 2 (Koin | Lingkaran) :
LBP : 11518.858406977664
COLOR : 31100.77087647685
SHAPE : 51.742306783209926
```

```
print("Berikut nilai euclidean untuk setiap feature extraction citra 1 (Piramida | Segitiga) : ")
print("LBP :",euclideanDistance(lbpSegitiga))
print("COLOR :",euclideanDistance(colorSegitiga))
print("SHAPE :",euclideanDistance(shapeSegitiga))
```

```
Berikut nilai euclidean untuk setiap feature extraction citra 1 (Piramida | Segitiga) :
LBP : 10996.311427019516
COLOR : 26816.40115468307
SHAPE : 36.42734424073011
```

Dari data tersebut kita dapat mengetahui bahwa :

1. **Color descriptor** memberikan hasil ekstraksi fitur yang banyak, dilihat dari nilai *Euclidean* dan plot histogramnya, hal ini terjadi dikarenakan, metode ini mengekstrak fitur berdasarkan warna yang ada pada citra dimana nilai pixelnya berarti berkisar antara 0 – 255, dan setiap pixel nya di ekstraksi untuk menyimpan fiturnya, jadi didapatkanlah fitur yang banyak dengan nilai yang besar, akan tetapi terdapat beberapa fitur yang tidak terlalu berguna seperti hasil ekstraksi fitur dari *background* yang ikut di ekstrak,. Biasanya untuk mengurangi komputasi pada komputer, para peneliti akan melakukan normalisasi agar nilai fitur yang masuk tidak terlalu besar.
2. **LBP** memberikan hasil ekstraksi fitur yang lebih sedikit dari *color descriptor*. Hal ini terjadi karena LBP melakukan penghitungan nilai pixel yang lebih banyak dari *color descriptor*. Maka dari itu LBP menghasilkan nilai ekstraksi yang bisa jadi lebih kecil, akan tetapi tetap mewakili representasi dari citra tersebut. Sama dengan *color descriptor*, LBP juga perlu dilakukan normalisasi agar komputasi pada computer bisa lebih ringan.
3. **Prewitt Transform** memberikan hasil ekstraksi fitur yang paling kecil diantara dua metode sebelumnya. Hal ini terjadi karena metode ini melakukan penghitungan pixel atau ekstraksi fitur berdasarkan garis yang membentuk objek pada citra tersebut atau lebih tepatnya berdasarkan bentuk objek yang terdapat didalam citra tersebut. Yang bisa dilihat contoh plot yang akan di ekstraksi oleh metode ini.



Jadi metode ini akan mengekstraksi bagian tepi yang membentuk objek pada citra tersebut, seperti bentuk tepi lingkaran, angka, huruf yang terdapat pada gambar tersebut dan nilainya berkisar -1 hingga 1 atau dari warna hitam ke putih. Hal ini yang membuat hasil ekstraksi fiturnya bernilai kecil.