

# Extração e Análise de Fundo em Imagens

Pedro Cunial Campos – Insper

Outubro 2018

## 1 Introdução

Na detecção e contagem de objetos em uma imagem é comum a existência de um pré-processamento da mesma, a fim de facilitar a detecção dos mesmos.

Apesar das diversas formas de pré-processamento que possam ser feitas, neste artigo pretende-se discutir a eficácia e implementação do método de *codebooks*[2] para extração de fundo (*background subtraction*).

## 2 Ferramental

Para a implementação do algoritmo[2], fora utilizado o *dataset* PETS 2009[3] (figura 1); Um conjunto de fotos tiradas sequencialmente em pontos constantes. O objetivo do uso do conjunto é a redução de fatores externos nos resultados do estudo.

Além disso, o algoritmo fora implementado em *Python*, com o auxílio da biblioteca *OpenCV*, tal como o *Numpy*.



Figure 1: Exemplo de imagem do *dataset* PETS 2009

### 3 Metodologia e Implementação

O método de *Codebooks*[2] consiste, principalmente, de duas partes:

---

Algorithm for Codebook construction

---

```

I.  $L \leftarrow 0^1$ ,  $\mathcal{C} \leftarrow \emptyset$  (empty set)
II. for  $t = 1$  to  $N$  do
    (i)  $\mathbf{x}_t = (R, G, B)$ ,  $I \leftarrow \sqrt{R^2 + G^2 + B^2}$ 
    (ii) Find the codeword  $\mathbf{c}_m$  in  $\mathcal{C} = \{\mathbf{c}_i | 1 \leq i \leq L\}$  matching to  $\mathbf{x}_t$  based on two conditions (a) and (b).
        (a)  $colordist(\mathbf{x}_t, \mathbf{v}_m) \leq \epsilon_1$ 
        (b)  $brightness(I, \langle \hat{I}_m, \hat{I}_m \rangle) = \text{true}$ 
    (iii) If  $\mathcal{C} = \emptyset$  or there is no match, then  $L \leftarrow L + 1$ . Create a new codeword  $\mathbf{c}_L$  by setting
        •  $\mathbf{v}_L \leftarrow (R, G, B)$ 
        •  $\mathbf{aux}_L \leftarrow \langle I, I, 1, t - 1, t, t \rangle$ .
    (iv) Otherwise, update the matched codeword  $\mathbf{c}_m$ , consisting of
         $\mathbf{v}_m = (\hat{R}_m, \hat{G}_m, \hat{B}_m)$  and  $\mathbf{aux}_m = \langle \hat{I}_m, \hat{I}_m, f_m, \lambda_m, p_m, q_m \rangle$ , by setting
        •  $\mathbf{v}_m \leftarrow \left( \frac{\hat{I}_m \hat{R}_m + R}{f_m + 1}, \frac{\hat{I}_m \hat{G}_m + G}{f_m + 1}, \frac{\hat{I}_m \hat{B}_m + B}{f_m + 1} \right)$ 
        •  $\mathbf{aux}_m \leftarrow \langle \min\{I, \hat{I}_m\}, \max\{I, \hat{I}_m\}, f_m + 1, \max\{\lambda_m, t - q_m\}, p_m, t \rangle$ .
    end for
III. For each codeword  $\mathbf{c}_i$ ,  $i = 1, \dots, L$ , wrap around  $\lambda_i$  by setting  $\lambda_i \leftarrow \max\{\lambda_i, (N - q_i + p_i - 1)\}$ .

```

---

Figure 2: Algoritmo de treinamento de *Codebooks*[2]

Primeiro o treinamento de *codebooks* segundo o algoritmo descrito pelos autores[2] (figura 2). Onde a função de brightness retorna um *booleano* baseado no  $I$  de um dado pixel pertencer à:

$$\alpha I_{max} \leq I \leq \min(\beta I_{max}, I_{min}/\alpha)$$

A função de *colordist* pelo resultado da figura 3.

Os valores de  $\alpha$ ,  $\beta$  e  $\epsilon_1$  são todos hiper-parâmetros do modelo e devem ser definidos pelo usuário.

Ao fim do treinamento dos *codebooks*, os seus valores são salvos no arquivo *test\_save.npy*, para que possam ser reutilizados sem a necessidade de um novo treinamento (o que é consideravelmente custoso em termos de tempo, levando cerca de 5 segundos por imagem no conjunto de treinamento).

Com isso, o segundo passo é a extração de fundo da imagem, segundo o algoritmo descrito na figura 4. Em suma, para uma imagem no mesmo local do conjunto de treinamento, mas não pertencente ao mesmo, ele confere se cada pixel poderia pertencer a um *codebook* conhecido; Se puder, o pixel é considerado como fundo da imagem, caso contrário é considerado como frente.

Finalmente, com o fundo e frente da imagem definidos, a contagem de pessoas na mesma foi realizada a partir da detecção de *blobs* cuja área seriam maiores do que um hiper-parâmetro (nomeado de *thresh*). Para a detecção destes *blobs*, utilizou-se a função de geração de contornos do próprio *OpenCV*, bem como a sua função para o cálculo da área dos mesmos[1].

$$\begin{aligned}
\|\mathbf{x}_t\|^2 &= R^2 + G^2 + B^2, \\
\|\mathbf{v}_i\|^2 &= \tilde{R}_i^2 + \tilde{G}_i^2 + \tilde{B}_i^2, \\
\langle \mathbf{x}_t, \mathbf{v}_i \rangle^2 &= (\tilde{R}_i R + \tilde{G}_i G + \tilde{B}_i B)^2.
\end{aligned}$$

The color distortion  $\delta$  can be calculated by

$$\begin{aligned}
p^2 &= \|\mathbf{x}_t\|^2 \cos^2 \theta = \frac{\langle \mathbf{x}_t, \mathbf{v}_i \rangle^2}{\|\mathbf{v}_i\|^2}, \\
\text{colordist}(\mathbf{x}_t, \mathbf{v}_i) &= \delta = \sqrt{\|\mathbf{x}_t\|^2 - p^2}.
\end{aligned}$$

Figure 3: Método para cálculo de *colordist*[2]

---

Algorithm for Background subtraction	
I.	$\mathbf{x} = (R, G, B), I \leftarrow \sqrt{R^2 + G^2 + B^2}$
II.	For all codewords in $\mathcal{M}$ in Eq. (1), find the codeword $\mathbf{c}_m$ matching to $\mathbf{x}$ based on two conditions: <ul style="list-style-type: none"> <li>• <math>\text{colordist}(\mathbf{x}, \mathbf{c}_m) \leq \epsilon_2</math></li> <li>• <math>\text{brightness}(I, \langle \hat{I}_m, \hat{I}_m \rangle) = \text{true}</math></li> </ul> Update the matched codeword as in Step II (iv) in the algorithm of codebook construction.
III.	$BGS(\mathbf{x}) = \begin{cases} \text{foreground} & \text{if there is no match} \\ \text{background} & \text{otherwise.} \end{cases}$

---

Figure 4: Algoritmo para extração de fundo[2]

## 4 Resultados

Aplicando o ferramental discutido, foi possível chegar na imagem segmentada demonstrada pela figura 6 a partir da imagem da figura 5, segundo os hiperparâmetros:

$$\alpha = 0.6$$

$$\beta = 1.1$$

$$\epsilon_1 = 12$$

$$\epsilon_2 = 50$$

$$\text{thresh} = 200$$

Com isso, foi possível contar 5 pessoas na imagem. No entanto, o modelo ainda possui algumas limitações, como por exemplo a detecção do poste como um membro da frente da imagem (o que, provavelmente, é devido ao reflexo da luz no mesmo variar em diferentes horas do dia).



Figure 5: Imagem inicial (não segmentada)

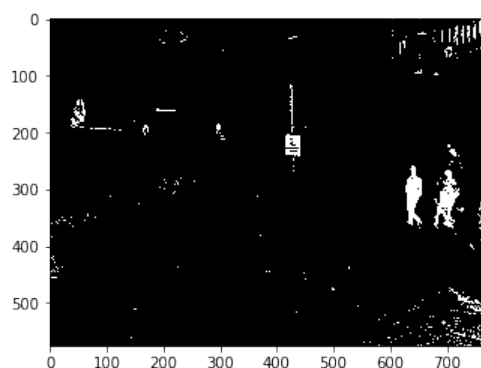


Figure 6: Imagem segmentada, resultado da aplicação do algoritmo de extração de fundo

## 5 Conclusões

Em suma, pode-se perceber a eficácia do método para a detecção de fundo de imagens. No entanto, limitações como as sugeridas pela detecção do poste como membro da frente sugerem que o modelo ainda tenha as suas falhas.

Para estudos futuros, aumentar espectro de tempo utilizado nas imagens do conjunto de treinamento pode ser uma boa hipótese para corrigir o problema.

Além disso, o estudo poderia explorar outros métodos de detecção e contagem de pessoas, bem como diferentes métodos para a extração de fundo da imagem (como o *MOG*, por exemplo).

## References

- [1] [https://docs.opencv.org/3.3.1/d4/d73/tutorial\\_py\\_contours\\_begin.html](https://docs.opencv.org/3.3.1/d4/d73/tutorial_py_contours_begin.html).

- [2] David Harwooda Larry Davisa Kyungnam Kima, Thanarat H. Chalid-abhongseb. *Real-time foreground-background segmentation using codebook model*. 2005.
- [3] PETS. Pets 2009 benchmark data. <http://www.cvg.reading.ac.uk/PETS2009/a.html>, 2009.