A Collaborative Filtering Based Movie Recommendation System

Computational Intelligence Lab, Spring 2018
April 22, 2018

PETROVA Kalina
Department of Computer Science, ETH Zurich, Switzerland
Email: dddxxx@ethz.ch

IVANOV Petar
Department of Computer Science, ETH Zurich, Switzerland
Email: ddd@lll.ethz.ch

LI Shiyi
Department of Earth Sciences, ETH Zurich, Switzerland
Email: lishi@student.ethz.ch

Abstract

This is the abstract

I. INTRODUCTION

Here is the introduction.

II. METHODS

Hello I am figures and algorithms

Input: Image I, mask M, kernel K and threshold ϵ
Result: Reconstructed image I_r
$K \leftarrow \frac{K}{\sum_i \sum_j K_{i,j}}$ (normalize K to preserve energy);
$I_{prev} \leftarrow 0_{size(I)}$;
$I_r \leftarrow I$;
while $\|I_r - I_{prev}\|_F > \epsilon$ **do**
    $I_{prev} \leftarrow I_r$;
    $I_r \leftarrow \text{convolve}(I_r, K)$;
    $I_r \leftarrow I_r \circ \mathbf{1}_{M=0} + I \circ \mathbf{1}_{M\neq 0}$;
end

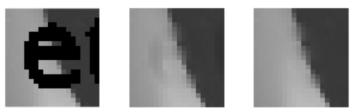
Algorithm 1: Diffusion algorithm for inpainting. We denote element-wise multiplication with the $\circ$ operator. The $\mathbf{1}_{M=0}$ function represents a matrix with elements (i, j) set to 1 when $M_{i,j} = 0$ and 0 otherwise.

1) *Constructing a directional kernel:* I am a subsubsection
2) *Per-patch diffusion using K_θ:* I am another subsubsection.

$$K_{\text{diamond}} = \begin{bmatrix} 0 & 0.25 & 0 \\ 0.25 & 0 & 0.25 \\ 0 & 0.25 & 0 \end{bmatrix}$$
$$K_{\text{diag}} = \begin{bmatrix} 0.38 & 0.04 & 0.04 \\ 0.04 & 0 & 0.04 \\ 0.04 & 0.04 & 0.38 \end{bmatrix}$$

Figure 1: Kernels used for diffusion.

(a) Diamond kernel K_{diamond}



(b) Directional kernel K_θ for $\theta = 100°$.

Figure 2: Step-by-step illustration of the diffusion process with different kernels. Each step represent 20 iterations.

Algorithm	MSE	Runtime
Directional Diffusion (16×16)	$\mathbf{0.00055} \pm 0.00051$	8.7 ± 2.13
Directional Diffusion (32×32)	0.00057 ± 0.00053	2.4 ± 0.04
Diffusion (K_{diamond})	0.00061 ± 0.00057	$\mathbf{0.5} \pm 0.06$
Sparse-coding (DCT)	0.0015 ± 0.0012	12.8 ± 4.67
Sparse-coding (Haar wavelet)	0.0024 ± 0.0021	13.0 ± 3.72
Singular Value Decomposition	0.0019 ± 0.0018	0.7 ± 0.09

Table I: Mean squared error and runtime (in seconds) across different algorithms for the text mask. The best result is highlighted in bold.

III. Results

This is results.

Example of list:

1) Sparse-coding with a DCT dictionary [1].
2) Sparse-coding with a Haar wavelet [2].
3) Singular Value Decomposition [3].
4) Regular diffusion with a K_{diamond} kernel.
5) Directional diffusion with patches of size 16×16.
6) Directional diffusion with patches of size 32×32.

blabblabla:

$$\text{MSE}(I, I^{\text{rec}}) = \frac{1}{512 \cdot 512} \sum_{i,j} (I_{i,j} - I^{\text{rec}}_{i,j})^2$$

Hello I am a table

IV. Discussion

why?

because sky is high!

V. Conclusion

U are close

References

[1] T. Hofmann, B. McWilliams, and J. Buhmann, "Sparse coding, overcomplete dictionaries," 2015, http://cil.inf.ethz.ch/material/lecture/lecture09.pdf.

[2] ——, "Sparse coding, classical methods," 2015, http://cil.inf.ethz.ch/material/lecture/lecture08.pdf.

[3] T. Hofmann, "Singular value decomposition," 2015, http://cil.inf.ethz.ch/material/lecture/lecture03.pdf.