Applied Mathematics and Computation 269 (2015) 464–472

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

A locally adaptive, diffusion based text binarization technique

B.A. Jacobs [a,b,*], E. Momoniat [a,b]

[a] School of Computer Science and Applied Mathematics, University of the Witswatersrand, Johannesburg, Private Bag 3, Wits 2050, South Africa
[b] DST-NRF Centre of Excellence in Mathematical and Statistical Sciences (CoE-MaSS), University of the Witwatersrand, Johannesburg, Private Bag 3, Wits 2050, South Africa

ARTICLE INFO

Keywords:
Binarization
Image denoising
Diffusion
Fitzhugh–Nagumo
Document image
GPGPU

ABSTRACT

This research proposes an adaptive modification to a novel diffusion based text binarization technique. This technique uses linear diffusion with a nonlinear source term to achieve a binarizing effect. This simple isotropic process is compared to the state-of-the-art DIBCO contestants and produces remarkable results given the simplicity of the algorithm. Furthermore, the authors show how using a simple discretization scheme allows for the massively parallel implementation of this process.

1. Introduction

Document image binarization is a process in which an input image is segmented so that the text is contained in one segment, represented by black, and the background information is contained in the other, represented as white. This has become an important preliminary step in document image analysis and allows optical character recognition (OCR) algorithms to work more effectively.

In the recent years document image binarization has received a substantial effort. A thorough survey of this work is presented in [1–4]. More recently there has been research done on the application of tensor based anisotropic diffusion processes [5–8]. These efforts include performing backward diffusion to undo the effects of ink diffusion on paper [7] which are typically associated with document degradation, as well as applying an anisotropic process to a low quality input image captured by a low cost imaging device or camera phone [8] to obtain a binary image. In stark contrast to these processes the process presented here is extremely simple, isotropic and still effective.

In broadly describing the binarization process we begin by representing our initial data as a two-dimensional matrix, or a height map, where the values at an (x, y) coordinate indicate the height of that pixel. These binarization techniques construct a threshold value such that if we slice the height map at this value we will segment the image in a meaningful way. That is to say, all pixels above the threshold will be classified as foreground and all those pixels below the threshold will be classified as background. Global methods construct this threshold based on information derived from the entire image.

Using a globally defined threshold leads to the inadequacy of thresholding every pixel according to irrelevant information. Locally adaptive methods improve upon this by selecting a threshold based on local information.

* Corresponding author at: School of Computer Science and Applied Mathematics, University of the Witswatersrand, Johannesburg, Private Bag 3, Wits 2050, South Africa.
E-mail addresses: byron@jsphere.com, byron.jacobs@wits.ac.za (B.A. Jacobs), ebrahim.momoniat@wits.ac.za (E. Momoniat).

http://dx.doi.org/10.1016/j.amc.2015.07.091
0096-3003/

The authors recently introduced [9] a novel approach to document image binarization based on a diffusion model with a nonlinear source term. The model can be written as

$$\frac{\partial u}{\partial t} = c_d\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) + c_s u(1-u)(u-a), \tag{1}$$

subject to

$$u(x, y, 0) = \mathrm{Image}(x, y), \tag{2}$$

and Neumann boundary conditions on the image boundary. Here c_d represents the coefficient of diffusion, c_s is the scaling coefficient of the source term, a is a thresholding parameter and $\mathrm{Image}(x, y)$ is the initial image data. The first term in Eq. (1) describes a diffusion process and the second describes a thresholding process. The benefits of this model are outlined in [9] but perhaps the reactive behavior of this process is the strongest asset. In [9] the authors chose the thresholding parameter a globally. In this paper we extend the model by selecting a based on local information making the process locally adaptive.

By localizing the process we are able to process each local sector independently. This means that this process is inherently parallel. Furthermore we discretize our model explicitly so that we may operate on the discrete input data such that each iteration of the method is dependant only on the immediately preceding iteration. Therefore the use of a General Purpose Graphics Processing Unit (GPGPU) arises naturally. GPGPU programming flourishes when an application is massively parallel in nature.

GPUs are becoming more and more a part of any computationally intensive research and the benefits of GPGPU programming are evident in the literature. Cruz et al. [10] give an introduction to GPU architecture and show the efficacy of this technique through application of two fast summation algorithms (fast Gaussian Transform and fast Multipole Method). Su and Xu [11] implement a wavelet-based image denoising algorithm on GPU hardware. The interested reader is directed to [12,13] for a description of the GPU architecture mechanics. By discretizing (1) we may reduce applying this PDE to an image to a series of linear algebraic operations. Krüger and Westermann [14] develop strategies for implementing these linear algebra operators on a GPU. Bolz et al. [15] implement an efficient sparse matrix solver using GPU architecture for solving linear systems, which arise in implicit discretization schemes. Typically large computational costs are involved in the transfer of image data to and from the GPU memory. Fortunately modern GPU hardware has enough memory to allow for a single transmission of data in each direction, without the need to swap in sections of the data set. This makes the problem of an isotropic, explicit scheme even more amenable to GPGPU programming.

In Section 2.1 we further substantiate our construction of this model by discussing the merits of a diffusion based model. We then go on to show how the model is localized in Section 2.2. A simple implementation of our method is presented and applications of GPGPU are highlighted in Section 3. The results in Section 5 illustrate the efficacy of our adaptive method. In the interest of brevity we choose not to compare our method with the extensive range of binarization techniques that exist in the literature. Instead we compute standard performance measures which are used in the Document Image Binarization Competition (DIBCO) series as a benchmark. This allows us to compare our method with the state-of-the-art methods without the need to compare our method with a plethora of new techniques. We do, however, provide some illustrative examples to indicate the performance of our method and subjectively compare with the seminal method of Sauvola [16]. We finally make some concluding remarks in Section 6.

2. Model derivation

2.1. Diffusion and a source term

The effect of image denoising has been well established in the literature and the interested reader is directed to [17–22]. The basis of our model is the well known diffusion equation in two-dimensions.

$$\frac{\partial u}{\partial t} = c_d\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right). \tag{3}$$

We affirm the use of a diffusion based equation by means of a simple example. Consider the small window of an image.

$$\begin{pmatrix} 0.82 & 0.93 & 0.87 \\ 0.90 & 0.49 & 0.95 \\ 0.93 & 0.95 & 0.88 \end{pmatrix}$$

If the threshold is determined to be 0.5 then the center pixel would be classified as background and the surrounding pixels as foreground. The diffusion model will circumvent this and draw the center pixel's value toward its neighbors, homogenizing the area and classifying the outlying center pixel as a foreground element. The reactive nature of this model was briefly alluded to in Section 1, but it is this dynamic behavior that empowers the method.

A consequence of diffusion is the blurring of information and while some homogeneity may assist in cementing the binarization process, too much may inhibit it. We therefore include the Fitzhugh–Nagumo inspired source term

$$c_s u(1-u)(u-a) \tag{4}$$