On the Efficiency of the Mean Intercept Length Tensor

Rodrigo Moreno∗†, Örjan Smedby∗† and Magnus Borga∗‡
∗Center for Medical Image Science and Visualization (CMIV), Linköping University
†Department of Medical and Health Sciences (IMH), Linköping University
‡Department of Biomedical Engineering (IMT), Linköping University
Linköping University, Campus US, 581 85, Linköping, Sweden
Email: {rodrigo.moreno,orjan.smedby,magnus.borga}@liu.se

Abstract—The Mean Intercept Length tensor is one of the most used techniques to estimate microstructure orientation and anisotropy of materials from 2D or 3D binary images. This paper proposes an efficient implementation of this technique. First, the Extended Gaussian Image is computed for the binary image. Second, the intercepts are computed for all possible orientations through an angular convolution. Finally, the tensor is computed by means of the covariance matrix. The complexity of the method is \( O(n + m) \) in contrast with \( O(nm) \) of traditional implementations, where \( n \) is the number of voxels in the image and \( m \) is the number of orientations used in the computations.

I. INTRODUCTION

Estimating fabric tensors is one of the most widely used techniques to characterize the microstructural architecture of materials. Fabric tensors are able to estimate both anisotropy and orientation of a material (usually referred to as phase) with respect to another one.

The Mean Intercept Length (MIL) tensor [1] is the most used technique to compute fabric tensor. The MIL with respect to a particular orientation is defined as the mean distance between a change from one phase to the other in such an orientation. This value is inversely proportional to the number of intercepts between a set of parallel lines and the interface between both phases (see Figure 1). In 2D, the MIL tensor is obtained by applying ellipse fitting algorithms to polar plots of the MIL, also known as rose diagrams. A more detailed description of the process for computing the MIL tensor is given in the next section.

Accurate estimations of the MIL tensor usually involve high computational costs when traditional implementations are applied. This makes it difficult to apply this technique to large scans. Thus, this paper aims at proposing an efficient implementation of the MIL tensor.

The paper is organized as follows. Section II presents a mathematical derivation of the MIL tensor in 2D and 3D. Section III shows some experiments in 2D and 3D. Finally, Section IV discusses the results and makes some final remarks.

II. THE MIL TENSOR

The following subsections describe both the traditional and the reformulation of the MIL tensor.

A. Traditional Methodology

The traditional methodology for computing the MIL can be summarized as follows. First, a family of parallel lines to a specified direction \( \mathbf{v} \) are traced, with \( \mathbf{v} \) being a unitary vector. Second, the number of intersections, \( C(\mathbf{v}) \), between these lines and the interface between both phases is counted (cf. Figure 1). Finally, the MIL with respect to \( \mathbf{v} \), \( \text{MIL}(\mathbf{v}) \), can be computed as:

\[
\text{MIL}(\mathbf{v}) = \frac{h}{C(\mathbf{v})},
\]

where \( h \) is the summation of the length of all traced lines.

In order to obtain the MIL tensor in 2D, the MIL for different orientations \( \mathbf{v} \) is plotted in a rose diagram. Researchers have found experimentally that the rose diagram of the MIL is usually close to an ellipse for many types of materials, in particular for trabecular bone [1]. This fact makes it possible to estimate with a small error the parameters of such an ellipse by any fitting algorithm. Thus, the MIL tensor, \( M \), can be computed as the \( 2 \times 2 \) matrix that represents the estimated ellipse.

In 3D, the process for computing the MIL tensor is similar. The difference in this case is that the MIL for different orientations should be fitted to an ellipsoid. The parameters of the resulting ellipsoid can be used to compute the MIL tensor, which is the \( 3 \times 3 \) matrix that represents such an ellipsoid.

The accuracy of the method can be controlled by increasing the number of traced lines and by using more directions \( \mathbf{v} \). A usual setting is to trace as many lines as needed to completely cover the image, that is \( h = n \), where \( n \) is the number of pixels (voxels in 3D). In this...
case, the total complexity of computing the MIL tensor through the methodology described above is $O(nm)$, where $m$ is the number of different directions of $v$, since the complexity of computing $C(v)$ is $O(n)$, which is mainly determined by the complexity of the utilized line-drawing method (e.g., the Bresenham’s algorithm).

### B. Reformulation of the MIL Tensor

This subsection describes a general formulation of the MIL tensor, which is equivalent to the traditional methodology described in the previous subsection. The technique can be reformulated as follows. First, consider the case of Figure 2. It shows a square region of interest where the interface between both phases is a straight line of length $l$ with a unitary normal $u$. The same figure also shows the traced lines for a specific direction $v$, which are separated from one another by a distance $\epsilon$. As described in [2], it is not difficult to show that:

$$C(v) = \frac{l}{\epsilon} t(v, -u), \quad (2)$$

with:

$$t(a, b) = \begin{cases} \langle a, b \rangle & \text{if } \langle a, b \rangle > 0 \\ 0 & \text{otherwise} \end{cases}, \quad (3)$$

where $\langle \cdot, \cdot \rangle$ is the inner product of vectors. Function $t$ is usually referred to as the half-cosine function [3].

It is important to remark that in general $C(v) \neq C(-v)$. For example, in Figure 2, $C(v) = 4$, but $C(-v) = 0$, since only bright to dark interfaces are taken into account. If a small enough value of $\epsilon$ is used, this result can be generalized to any shape of the interface between phases in 2D as:

$$C(v) = \frac{1}{\epsilon} \int_L t(v, -\hat{s}) \, dL, \quad (4)$$

where $L$ is the interface between phases and $\hat{s}$ is the normal at every point of the interface. Note that $L$ can consist of a set of different contours. In addition, $\epsilon$ can be thought of as a parameter that should be adjusted in order to appropriately sample $L$. This equation can be rewritten as:

$$C(v) = \frac{1}{\epsilon} \int_L \int_0^{2\pi} t(v, -u_\theta) \delta(-u_\theta - \hat{s}) \, d\theta \, dL, \quad (5)$$

where $\delta$ is the unit impulse function, and $u_\theta = (\cos(\theta) \sin(\theta))^T$. This integral can be simplified with the help of the Extended Gaussian Image (EGI) [4], which can be considered as the angular histogram of normals [5]. The EGI at a specific direction $u_\theta$, $G(u_\theta)$, is given by:

$$G(u_\theta) = \int_L \delta(u_\theta - \hat{s}) \, dL. \quad (6)$$

Thus, by changing the order of integration in (5), it is obtained that:

$$C(v) = \frac{1}{\epsilon} \int_0^{2\pi} \int_0^{2\pi} t(v, -u_\theta) G(-u_\theta) \, d\theta. \quad (7)$$

This equation shows that $C$ is equivalent to the angular convolution [6] between the mirrored EGI and the smoothing half-cosine kernel $t$. This equation can be used to estimate the MIL tensor, $M$. As already mentioned, $M$, is usually computed by means of ellipse fitting of the MIL($v$) for different values of $v$. Alternatively, since $M$ is proportional to the covariance matrix, it can be computed as:

$$M = w \int_0^{2\pi} \frac{v_\theta v_\theta^T}{C(v_\theta v_\theta^T)} \, d\theta \quad (8)$$

where $v_\theta = (\cos(\theta) \sin(\theta))^T$ and $w$ is a constant.

It is important to remark two aspects from (8). First, the size of $M$ is usually discarded, since it directly depends on the selection of $\epsilon$, which is a parameter of the method that is unrelated to the microstructural architecture of the studied material. Thus, analyses for estimating orientation and anisotropy from the MIL tensor are usually conducted on normalized versions of $M$ [7]. That means that, in practice, $w$ and $\epsilon$ can be safely omitted from the formulation, since they are removed in the normalization process of the MIL tensor. The second aspect to remark is that $M$ can only be computed if $C(v_\theta) > 0$ for all $v_\theta$. This condition can be assured by considering $L$ as a set of closed contours.

The extension of this formulation of the MIL tensor to 3D is straightforward. In this case:

$$C(v) = \frac{1}{\epsilon^2} \int_S t(v, -\hat{s}) \, dS \quad (9)$$

where $S$ is the interface surface between phases. The EGI in 3D is given by:

$$G(u_{\theta, \phi}) = \int_S \delta(u_{\theta, \phi} - \hat{s}) \, dS. \quad (10)$$

where $u_{\theta, \phi}$ is a vector in the unitary sphere given in spherical coordinates. Consequently:

$$C(v) = \frac{1}{\epsilon^2} \int_0^\pi \int_0^{2\pi} t(v, -u_{\theta, \phi}) G(-u_{\theta, \phi}) \sin(\phi) \, d\theta \, d\phi, \quad (11)$$

where $\sin(\phi) \, d\theta \, d\phi$ corresponds to the surface element of the unitary sphere in spherical coordinates. Similarly to the 2D case, (11) is equivalent to the angular convolution in spherical coordinates between the mirrored EGI and the smoothing half-cosine kernel $t$ in 3D. Finally:

$$M = w \int_0^\pi \int_0^{2\pi} \frac{v_{\theta, \phi} v_{\theta, \phi}^T}{C(v_{\theta, \phi} v_{\theta, \phi}^T)} \sin(\phi) \, d\theta \, d\phi, \quad (12)$$

with $v_{\theta, \phi}$ being a vector in the unitary sphere given in spherical coordinates, and $\sin(\phi) \, d\theta \, d\phi$ being the surface normal.
element of the unitary sphere in spherical coordinates. Following a similar reasoning, \( w \) and \( \epsilon \) can be safely omitted from the formulation.

**C. Efficiency**

The formulation presented in the previous subsection leads to a more efficient implementation of the MIL tensor than that obtained with traditional methodology. The computation of the MIL tensor can be achieved by dividing the process into three steps. In the first step, the EGI is computed for all possible directions. This can be done by computing the histogram of normals at the interface between phases. This step has a complexity of \( O(n) \) where \( n \) is the number of pixels (voxels).

The second step corresponds to the angular convolution computed through (7) in 2D, or (11) in 3D. Since the half-cosine is a smooth function, except in the discontinuity at 0 and \( \pi \), and has a large extent in the angular domain, it is more convenient to apply the angular convolution in the frequency domain. In 2D, this is done in the Fourier domain, at a maximum cost of \( O(m \log(m)) \), where \( m \) is the number of sampling orientations used in the computations. However, since the bandwidth of the Fourier transform of the half-cosine function is very narrow, the convolution can be approximated by considering just a few coefficients in the Fourier domain. For example, the introduced error in convolutions with the half-cosine function by truncating frequencies beyond \( \pi/32 \) in the Fourier domain is negligible. Thus, the complexity of the convolution is reduced to \( O(m) \) by using this approximation.

In 3D, the angular convolution is performed by using spherical harmonics [6]. In this case, the angular convolution has a complexity of \( O(N \log^2(N)) \), where \( N \) is the maximum order of the spherical harmonics [8]. Similarly to the 2D case, this complexity can be largely reduced, thanks to the properties of the half-cosine function. For example, the introduced error in convolutions with the half-cosine function by using at most 10-th order spherical harmonics is negligible [3]. Thus, the complexity is reduced to \( O(m) \) by using this approximation, where \( m \) is the number of orientations used in the computations.

In a third step, \( M \) is computed through (8) in 2D, or (12) in 3D, by using the precomputed values of the EGI. This step has also a complexity of \( O(m) \). Thus, the total complexity of the new formulation is \( O(n+m) \) compared to \( O(nm) \) of the traditional method.

**III. Experiments**

Figure 3a-d show the images used in the experiments in 2D. Figure 3a and 3b are synthetic blurred images of a bar and a cross contaminated with Gaussian noise, and Figures 3c and 3d correspond to regions of interest extracted from images acquired through Micro Computed Tomography (\( \mu \)CT) of specimens from the radius and a vertebra respectively. The synthetic images have been generated by adding Gaussian noise with zero mean and a standard deviation of 10% of the maximum gray value of every image. In turn, the tested images of the radius and vertebra correspond to slices of the images used in the experiments in 3D. A Gaussian filter with a standard deviation of 0.5 pixels has been applied to the thresholded images before computing the gradient in order to get more reliable estimations of the normals. Figures 3e-h show the ellipses that represent the MIL tensors computed for Figures 3a-d respectively. The MIL tensor is close to a rod aligned with the bar of Figure 3a, while the eigenvalues of the tensor for the cross of Figure 3b are proportional to its height and width. Orientation and anisotropy of trabecular bone can be obtained from the MIL tensor of the \( \mu \)CT images.

Figure 4a-d show the images used in the experiments in 3D. Figure 4a and 4b are renderings of two blurred synthetic images with some cylinders and walls respec-
tively, and Figure 4c and 4d correspond to µCT images of radius and vertebra specimens respectively. The cylinders and the walls have been contaminated with noise in the experiments. These noisy images have been generated by adding Gaussian noise with zero mean and a standard deviation of 10% of the maximum gray value of the image. Acquired images have an isotropic resolution of 20µm, 82µm for the µCT images of the radius and vertebra respectively. An isotropic resolution of 20µm has been used for the synthetic images. The MIL tensor has been computed using the same methodology described for 2D images.

Figure 4e-h show the ellipsoids that represent the MIL tensor for Figures 4a-d respectively. As in the case of 2D, the computed tensors for the synthetic images are in accordance with the expected results, that is, the MIL tensor is close to a rod aligned with the cylinders for Figure 4a, and close to a plate parallel to the walls for Figure 4b. It is important to remark that the MIL tensor does not depend on the number of cylinders or walls for these synthetic images, since the net effect of increasing the number of structures in these images is a scaling of the EGI which is eliminated when the tensor is normalized (cf. Subsection II-B). Results on µCT show that trabecular bone is anisotropic. This result is in accordance with previous works in trabecular bone [9], [10].

IV. DISCUSSION

This paper has presented an efficient method to compute the MIL tensor. The traditional procedure for computing the MIL tensor has been reformulated into more general equations. The new formulation consists of three independent steps that can be applied sequentially, namely the computation of the EGI, an angular convolution, and the covariance matrix.

The proposed formulation of the MIL tensor has mainly two advantages. On the one hand, it is less computationally expensive: $O(n + m)$ in contrast with $O(nm)$ of the traditional procedure. On the other hand, only the first step, that is the computation of the EGI, is directly applied to the image. This property can be used to extend the MIL tensor to gray-scale images by using a gray-scale variant of the EGI, such as the one proposed in [5]. Ongoing research includes the evaluation of this alternative for computing the MIL tensor in gray-scale.

V. ACKNOWLEDGEMENTS

We thank Prof. Osman Ratib from the Service of Nuclear Medicine at the Geneva University Hospitals for providing the µCT data of the vertebra and Andres Laib from SCANCO Medical AG and Torkel Brismar from the Division of Radiology at the Karolinska University Hospital for providing the µCT data of the radius. This research has been supported by the Swedish Research Council (VR), grant no. 2006-5670.

REFERENCES


