Comments on "A Closed-Form Solution to Tensor Voting: Theory and Applications"

Emmanuel Maggiori, Pablo Lotito, Hugo Luis Manterola and Mariana del Fresno

Abstract—We comment on a paper that describes a closed-form formulation to Tensor Voting, a technique to perceptually group clouds of points, usually applied to infer features in images. The authors proved an analytic solution to the technique, a highly relevant contribution considering that the original formulation required numerical integration, a time-consuming task. Their work constitutes the first closed-form expression for the Tensor Voting framework.

In this work we first observe that the proposed formulation leads to unexpected results which do not satisfy the constraints for a Tensor Voting output, hence they cannot be interpreted. Given that the closed-form expression is said to be an analytic equivalent solution, unexpected outputs should not be encountered unless there are flaws in the proof. We analyzed the underlying math to find which were the causes of these unexpected results.

In this commentary we show that their proposal does not in fact provide a proper analytic solution to Tensor Voting and we indicate the flaws in the proof.

Index Terms—Tensor Voting, perceptual grouping, feature inference.

1 Introduction

Tensor voting is a robust technique for perceptual grouping in noisy images. For a comprehensive source of information on the matter, we refer the reader to [1].

One of the main drawbacks of the technique is that it requires numerical integration, making execution times prohibitive in some contexts and mathematics complicated [2], [3]. The authors of [3] proposed in their work entitled "A closed-form solution to Tensor Voting: Theory and Applications" (CFTV) an analytic solution that does not require numerical integration and can be used for any number of dimensions and for any generic second-order symmetric tensors as an input.

This is highly relevant considering that over fifteen years passed from the original publication of the technique, and there had been no success at finding a closed-form solution to the problem. A number of reformulations have also been recently presented [2], [4], [5], though none of them features the simplicity of CFTV nor provides an analytic general solution.

- *Emmanuel Maggiori, Pladema, Facultad de Ciencias Exactas, Universidad Nacional del Centro de la Provincia de Buenos Aires, Tandil, Argentina.*
E-mail: emmanuelmaggiori@gmail.com
- *Pablo Lotito, Pladema, Facultad de Ciencias Exactas, Universidad Nacional del Centro de la Provincia de Buenos Aires and CONICET, Tandil, Argentina.*
E-mail: plotito@exa.unicen.edu.ar
- *Hugo Luis Manterola , Pladema, Facultad de Ciencias Exactas, Universidad Nacional del Centro de la Provincia de Buenos Aires and CONICET, Tandil, Argentina.*
E-mail: manterolaluis@gmail.com
- *Mariana del Fresno, Pladema, Facultad de Ciencias Exactas, Universidad Nacional del Centro de la Provincia de Buenos Aires and CIC-PBA, Tandil, Argentina.*
E-mail: mdelfres@gmail.com

In Tensor Voting, every input token, which is encoded as a second-order symmetric tensor, casts votes throughout the space. The field produced by a *stick* tensor, i.e. a tensor with only one non-zero eigenvalue, is computed analytically, and the other fields are derived after it. This is done by integrating the fields of a rotating *stick*.

In this communication, we show that CFTV does not in fact provide a solution to Tensor Voting. A counterexample is shown and the deviations from the expected results are illustrated. In addition, flaws in the proof are pointed out.

2 The Voting Fields in CFTV

The original fundamental *stick* field penalizes curvature and arc-length along the osculating circle from the voter to the receiver. In CFTV, the authors reformulated the fundamental *stick* field prior to deriving a closed-form solution. Curvature was replaced by a squared sine term, as in [4], [5], and distance is used instead of arc-length, as in [4].

Let $\mathbf{K}_j$ at $\mathbf{x}_j$ be a second-order symmetric *stick* tensor whose main eigenvector is $\mathbf{n}_j$ and τ_j the corresponding eigenvalue. This kind of tensor is usually used to represent the normal direction to an underlying feature in an image.

We wish to compute the vote $\mathbf{K}_{ij}$ cast by $\mathbf{K}_j$ at a given position $\mathbf{x}_i$. $\mathbf{r}_{ij}$ is a unit vector pointing from $\mathbf{x}_j$ to $\mathbf{x}_i$. The vote is defined as follows:

$$\mathbf{K}_{ij} = \mathbf{v}(\mathbf{x}_i, \mathbf{x}_j)\mathbf{v}(\mathbf{x}_i, \mathbf{x}_j)^T \eta(\mathbf{x}_i, \mathbf{x}_j, \mathbf{n}_j), \quad (1)$$

where $\mathbf{v}(\mathbf{x}_i, \mathbf{x}_j)$ is a function used to construct the resulting tensorial vote and $\eta(\mathbf{x}_i, \mathbf{x}_j, \mathbf{n}_j)$ is a decay factor.

Let's first analyze the decay factor, which is defined as follows:

$$exp\left(-\frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{\sigma_d}\right)(1 - (\mathbf{r}_{ij}^T \mathbf{n}_j)^2). \quad (2)$$

Votes are penalized with distance and curvature: the first through a Gaussian decay (with a range parameter σ_d) and the latter through the second factor of the equation, which is a squared sine.

Many possibilities exist for defining $\mathbf{v}(\mathbf{x}_i, \mathbf{x}_j)$. The authors, as well as the literature on Tensor Voting in general, chose the unique osculating circle which is normal to $\mathbf{n}_j$ and passes through $\mathbf{x}_i$ as the smooth continuation of the feature encoded in the voter. $\mathbf{v}(\mathbf{x}_i, \mathbf{x}_j)$ should return a vector that lies on the radius of the osculating circle. This function is defined as follows:

$$\mathbf{v}(\mathbf{x}_i, \mathbf{x}_j) = \tau_j \mathbf{n}_j - 2\mathbf{r}_{ij}^T \mathbf{n}_j \tau_j \mathbf{r}_{ij} = (\mathbf{n}_j - 2\mathbf{r}_{ij}(\mathbf{r}_{ij}^T \mathbf{n}_j))\tau_j. \quad (3)$$

Now we wish to define the vote cast by a generic tensor (not just a *stick* tensor). Let $\theta = (\phi_1, \phi_2, ..., \phi_{n-1})$ be the rotation angle in an n-dimensional space. The authors expressed the vote $\mathbf{S}_{ij}$ cast by a generic tensor in an integral that adds the different $\mathbf{K}_{\theta ij}$ produced by a *stick* that rotates around the complete set ν of unit normals encoded into the voter:

$$\mathbf{S}_{ij} = \int_{N_{\theta j} \in \nu} \mathbf{v}_\theta(x_i, x_j)\mathbf{v}_\theta(x_i, x_j)^T \eta(\mathbf{x}_i, \mathbf{x}_j, n_{\theta j}) d\mathbf{N}_{\theta j}. \quad (4)$$

In the integral, the different factors are subscripted with a θ to indicate the corresponding angle. In the original paper it is explicitly pointed out that $\int_{\mathbf{N}_{\theta j} \in \nu} \cdots d\mathbf{N}_{\theta j}$ is equivalent to $\int_{\phi_1}\int_{\phi_2} \cdots \int_{\phi_{n-1}} \cdots d\phi_{n-1} d\phi_{n-2} \cdots d\phi_1$ though in a simplified notation.

An analytic solution to (4) would constitute a closed-form solution to Tensor Voting.

Prior to addressing the formulation proposed in [3], we will first particularly remark the symmetry property of the matrices outputted by (4). The integrand of (4) is a symmetric matrix given that it consists of an outer product multiplied by a scalar. The integration of symmetric matrices must lead to a symmetric matrix too, hence it is expected that the application of Equation 4 on any input will be a symmetric tensor.

3 Closed-Form solution

The authors of [3] provided the following closed-form solution to Equation 4:

$$\mathbf{S}_{ij} = c_{ij}\mathbf{R}\mathbf{K}_j(\mathbf{I} - \frac{1}{2}\mathbf{r}\mathbf{r}^T)\mathbf{R}^T \quad (5)$$

$$\mathbf{R} = \mathbf{I} - 2\mathbf{r}\mathbf{r}^T \quad (6)$$

where the subscript notation has been simplified and c_{ij} is the Gaussian decay on the distance. In the original paper, the authors state that (4) and (5) are equivalent. In fact, it is explicitly stated that the votes computed after (5) are symmetric tensors, given the aforementioned equivalence of both expressions.

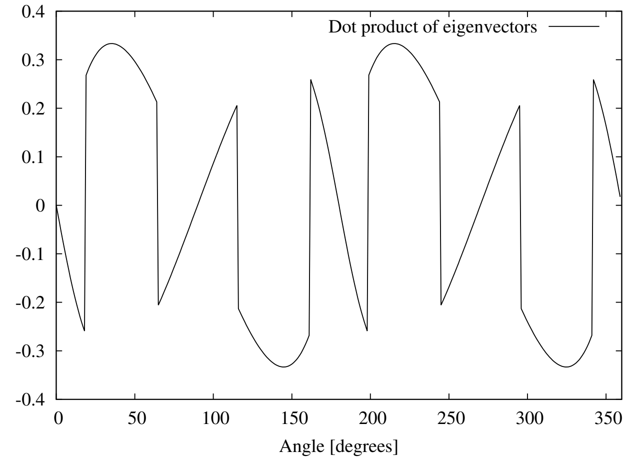

Fig. 1. Dot product of eigenvectors in 2D using CFTV. They are not orthogonal as expected.

We will here show a counterexample. Even though one counterexample is enough, we must point out that in most cases this unexpected behavior is observed. Suppose, for instance, that the tensor $\mathbf{K} = \begin{bmatrix} 1/2 & 0 \\ 0 & 1 \end{bmatrix}$ casts a vote in a 45^o direction, i.e $\mathbf{r} = \left[\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right]^T$. Following (5),

$$\mathbf{S}_{ij} = c_{ij}(\mathbf{I} - 2\mathbf{r}\mathbf{r}^T)\begin{bmatrix} 1/2 & 0 \\ 0 & 1 \end{bmatrix}(\mathbf{I} - \frac{1}{2}\mathbf{r}\mathbf{r}^T)(\mathbf{I} - 2\mathbf{r}\mathbf{r}^T)^T,$$

with $\mathbf{r}\mathbf{r}^T = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix}$, $\mathbf{S}_{ij} = c_{ij}\begin{bmatrix} 3/4 & -1/4 \\ -1/8 & 3/8 \end{bmatrix}$.

Considering that the result is not a symmetric matrix, (4) and (5) cannot be equivalent, hence the proof in [3] must be flawed and (5) does not constitute a closed-form solution to (4).

Given that the resulting matrix is not symmetric, the eigenvectors of the tensor are not guaranteed to be orthogonal. Tensor Voting encodes data in symmetric tensors and interprets data out of symmetric tensors. In 2D, for instance, the lowest eigenvalue is used as a measure of *junctionness*, because it encodes a component that contradicts the main direction. This only makes sense if the eigenvectors are orthogonal.

We tested the dot product of the eigenvectors when applying CFTV in 2D, which should equal zero if the votes are symmetric tensors. In Fig. 1 a plot of the vote cast by $\mathbf{K} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ is shown, which was computed for 360 directions ranging from 0 to 2π ignoring the euclidean distance scalar decay c_{ij}. An unexpected behavior is observed, resulting in tensors that are not orthogonal, with angles attaining up to 20^o of deviation from a right angle.

Let us see an example of the expected result and the outcome of CFTV. The vote cast by the *stick* $\mathbf{K} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ at $\mathbf{r} = \left[\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right]^T$ should be of the form

$\alpha \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$, i.e another stick rotated 90^o [1]. The result of applying (5) is, however, $c_{ij} \begin{bmatrix} 0 & 0 \\ -0.25 & 0.75 \end{bmatrix}$, which in addition to not being symmetric does not relate to the expected output. We are not aware of how the authors handled these situations throughout the rest of the paper.

4 The Proof of CFTV

First of all we will comment on the usage of $\mathbf{N}_\theta$ and $d\mathbf{N}_\theta$ throughout the proof. In a step the authors solve an integral by parts, in which they require to find a function whose derivative is $\mathbf{r}\mathbf{r}^T\mathbf{N}_\theta$. The function they found is $\frac{1}{2}\mathbf{r}\mathbf{r}^T\mathbf{N}_\theta^2$ implying that they considered

$$\frac{d}{d\mathbf{N}_\theta}[\mathbf{N}_\theta^2] = 2\mathbf{N}_\theta. \tag{7}$$

as following the standard rule of derivatives (something done in other parts of the proof too). At another stage they convert the expression $\frac{d}{d\mathbf{N}_\theta}[\mathbf{r}\mathbf{r}^T\mathbf{N}_\theta^2]$ into $\frac{d}{d\mathbf{N}_\theta}[\mathbf{r}\mathbf{r}^T\mathbf{N}_\theta]$. This is because of the property $\mathbf{N}_\theta^k = \mathbf{N}_\theta$ that stands for $k \in \mathbb{Z}^+$, which is true observing that $\mathbf{N}_\theta^k = \mathbf{n}_\theta\mathbf{n}_\theta^T\mathbf{n}_\theta\mathbf{n}_\theta^T \ldots \mathbf{n}_\theta\mathbf{n}_\theta^T = \mathbf{n}_\theta \cdot 1 \ldots 1 \cdot \mathbf{n}_\theta^T = \mathbf{n}_\theta\mathbf{n}_\theta^T = \mathbf{N}_\theta$. However, following the same principles it must also be true that $\frac{d}{d\mathbf{N}_\theta}[\mathbf{N}_\theta^2] = \frac{d}{d\mathbf{N}_\theta}[\mathbf{N}_\theta] = \mathbf{I}$, which contradicts (7). This shows that integrating with respect to that matrix as if it were a variable in a polynomial is not correct.

In addition, there is no apparent reason why the authors should be integrating and differentiating with respect to the matrix $\mathbf{N}_\theta$. As stated before, $\int_{\phi_1}\int_{\phi_2}\cdots\int_{\phi_{n-1}}\cdots d\phi_{n-1}d\phi_{n-2}\cdots d\phi_1$ was simplified to $\int_{\mathbf{N}_{\theta j}\in\nu}\cdots d\mathbf{N}_{\theta j}$. In the two-dimensional case, for example, the actual integration would be of the form $\int_0^{2\pi}\ldots d\phi_1$ or in three dimensions $\int_0^{2\pi}\int_0^{2\pi}\ldots d\phi_2 d\phi_1$. It would have been probably better to use $d\theta$ instead. The integral should be computed with respect to the various rotation angles of the *stick* and not with respect to a matrix, as was the original intention prior to simplifying the notation. Besides, it is not clear why in the case of integrating with respect to a matrix, a matrix power would be used instead of an element-wise operation.

There is another concern with the proof. At a certain point the authors compute $\frac{d}{d\mathbf{N}_\theta}(\tau_\theta^2\mathbf{N}_\theta)$ as $\tau_\theta^2\mathbf{I}$, which implies that the length τ_θ was considered constant. Even in the case that differentiating with respect to $\mathbf{N}_\theta$ would have been correct, the length τ_θ would not be necessarily constant. τ_θ varies depending on the the angle θ. Given that $\mathbf{N}_\theta$ is a function of θ, τ_θ cannot be considered independent of $\mathbf{N}_\theta$. The only case in which that would be true is in the case of *ball* tensors, which can be decomposed into a constant length rotating *stick*. However, this is not true in the general case.

5 Conclusions

It has been shown that CFTV does not provide a proper analytic solution to Tensor Voting. This invalidates that (5) provides a closed-form solution as well as the subsequent proofs that were done upon that formulation.

The proof of a closed-form solution to Tensor Voting would have been of high impact, considering that the major concerns with the technique have to do with its formulation and applicability rather than to its robustness. The problem remains then unsolved.

As future directions of work, we suggest either to find the correct analytic solution to (4) or to further adapt the original *stick* formulation while preserving its perceptual spirit in order to facilitate the derivation of a closed-form solution.

References

[1] G. Medioni, M. Lee, and C. Tang, *A computational framework for feature extraction and segmentation*. Elsevier, 2000, ch. 4.
[2] P. Mordohai and G. Medioni, "Dimensionality estimation, manifold learning and function approximation using tensor voting," *Journal of Machine Learning Research*, vol. 11, pp. 411–450, 2010.
[3] T.-P. Wu, S.-K. Yeung, J. Jia, C.-K. Tang, and G. Medioni, "A closed-form solution to tensor voting: Theory and applications," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 8, pp. 1482–1495, 2012.
[4] R. Moreno, M. Garcia, D. Puig, L. Pizarro, B. Burgeth, and J. Weickert, "On improving the efficiency of tensor voting," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 11, pp. 2215–2228, 2011.
[5] E. Franken, M. Almsick, P. Rongen, L. Florack, and B. Haar Romeny, "An efficient method for tensor voting using steerable filters," in *Computer Vision - ECCV 2006*, ser. Lecture Notes in Computer Science, A. Leonardis, H. Bischof, and A. Pinz, Eds. Springer Berlin Heidelberg, 2006, vol. 3954, pp. 228–240.