ASEE 2022 ANNUAL CONFERENCE
Excellence Through Diversity
MINNEAPOLIS, MINNESOTA, JUNE 26TH-29TH, 2022 ASEE

Paper ID #36438

Visualizing tensor component transformations using virtual reality and web-based applications

John Sanders

John W. Sanders is currently an Assistant Professor of Mechanical Engineering at California State University, Fullerton. He holds a Ph.D. and M.S. in Theoretical and Applied Mechanics from the University of Illinois at Urbana-Champaign, and a B.S. in Engineering Physics and Mathematics from Saint Louis University. His research interests include clean energy, fracture mechanics, nonlinear dynamics, and STEM education research.

Serop Kelkelian

Serop Kelkelian is currently a senior undergraduate at California State University, Fullerton majoring in Mechanical Engineering and minoring in Computer Science. Presently, he is completing a summer internship with the United States Department of Defense. After graduation, he plans to pursue a graduate degree in Computer Science with a focus on artificial intelligence.

Markus Wieser

Markus Wieser obtained his B.Sc. degree in Automotive Engineering at the University of Applied Sciences Joanneum Graz. Currently, he is completing his M.Sc. studies at the same faculty.

Guenter Bischof (Associate Professor)

Günter Bischof holds a doctorate in physics and is currently an Associate Professor of Applied Mathematics at the University of Applied Sciences FH Joanneum in Graz.

Visualizing tensor component transformations using virtual reality and web-based applications

John W. Sanders[a] Serop Kelkelian[a] Markus Wieser[b] Günter Bischof[b]

[a]*Department of Mechanical Engineering, California State University, Fullerton, United States*
[b]*Department of Automotive Engineering, University of Applied Sciences FH JOANNEUM, Austria*

Abstract

Tensors of the second rank, such as stress, strain, and the inertia tensor, are of fundamental importance in structural analysis and many other engineering applications. Unfortunately, the way in which these tensor components transform under coordinate rotations can be difficult to visualize and comprehend, and this poses a major conceptual challenge for many students. Mohr's circle is a graphical method commonly used to visualize planar stress transformations in traditional solid mechanics courses, but it has several drawbacks, including that it only applies to rotations about a single axis and that the angle subtended on the circle is not the actual angle of rotation. More recently, tensor component transformations have been illustrated in three dimensions with the aid of computer software. Currently these programs are static, in that the user specifies the initial tensor components and the rotation to be applied, and the program displays the final results without any intermediate history. In an effort to make these programs more engaging for students, the present authors have developed two pedagogical tools that illustrate three-dimensional tensor transformations dynamically, in real time: one using virtual reality software, the other using traditional web-based software. Both applications were created using the Unity game engine. In each case, the user manually manipulates a given system using either the hand controller (in a VR headset), the cursor (on a traditional computer), or their finger (on a mobile device), and the relevant tensor components update continuously while the transformations are being performed. All rotations are handled using quaternions in order to avoid gimbal lock. Both apps are available online completely free of charge for anyone to use. Here we give a detailed account of the development of these applications and the underlying theory.

1 Introduction and background

The pedagogical question of what tensors are [1–3] and how one ought to understand them—at once both philosophical and pragmatic—persists to the present day, due both to their mathematical nuance and to the prevalence of two competing schools of thought regarding their definition. The first approach to emerge for tensors of the second rank, sometimes referred to as the algebraic approach or the component approach, views tensors as sets of components that transform in prescribed ways under given coordinate transformations [4–8]. In general, a tensor of integer rank $r \geq 0$ may be defined as a set of N^r components $a_{i_1 i_2 \cdots i_r}$ (each index i_x ranges from 1 to N, where N is the dimension of the space of interest) that transform according to the following rule:

$$a'_{i_1 i_2 \cdots i_r} = R_{i_1 j_1} R_{i_2 j_2} \cdots R_{i_r j_r} a_{j_1 j_2 \cdots j_r}, \tag{1}$$

where the $a_{j_1 j_2 \cdots j_r}$ are the tensor's components in some coordinate frame S, the $a'_{i_1 i_2 \cdots i_r}$ are the components of the same tensor in a frame S' obtained from S by applying one or more of the transformations of interest (those being translations and proper rigid rotations in Euclidean space, and Lorentz transformations in Minkowski spacetime), R_{ij} is an orthogonal matrix representing the transformation, and we are employing the Einstein summation convention whereby repeated indices are implicitly summed over from 1 to N. The issue of covariance and contravariance [2] arises in non-Euclidean spaces where the metric tensor does not coincide with the identity [9].

The second approach, referred to formally as the geometric approach (and playfully by Misner, Thorne, and Wheeler [9] as the "machine-with-slots definition") views tensors as geometric objects with certain coordinate-independent properties [9–11]. According to the geometric approach, a vector may be defined as a quantity with a scalar magnitude and a direction in space, and higher-rank tensors may be defined as linear operators ("machines") that can operate on other tensors (by inserting them into its "slots"). In general, when a tensor of rank $r > 1$ operates on a tensor of rank $p < r$, the result is a new tensor of rank $r - p$.

The algebraic and geometric approaches are mathematically equivalent in that they yield the same computational results and it is straightforward to deduce the transformation rule (1) from the geometric definition [3]. It is perhaps a reflection of human psychology, then, that the academic community has become so staunchly divided between the two approaches. Indeed, the rift dates back to the formulation of relativity theory. According to Norton [12],

> In Einstein's hands, Lorentz covariance was a purely algebraic property. Space and time coordinates were, in effect, variables that transformed according to certain formulae. Hermann Minkowski was responsible for introducing geometric methods and thinking into relativity theory. He explained the background to his approach in his more popular (1909) lecture. [...] The difference between Einstein and Minkowski's approach to the same theory and even the same formalism is a polarity that will persist in various manifestations throughout the whole development of relativity theory, both special and general. Einstein's emphasis is on the algebraic properties of the theory, the equations that express its laws and their behaviour under transformation, its *covariance*. [...] Minkowski's emphasis is on the geometric properties of the theory, on those geometric entities which remain unchanged behind the transformations, its *invariance*. [12]

Evidently not everyone was impressed by Minkowski's geometrization. In his well known treatise on analytical mechanics, Lanczos [13] remarks rather cryptically,

> Little can be gained and a great deal lost in clarity if we try to operate with the tensor as a whole rather than its components. [13]

We can only speculate as to what Lanczos [13] meant by this. Perhaps he was referring to what Misner, Thorne, and Wheeler [9] describe as the "ambiguity of slots":

> Because the frame-independent geometric notation is somewhat ambiguous (which slots are being contracted? on which slot is the divergence taken? which slots are being transposed?), one often uses component notation to express coordinate-independent, geometric relations between geometric objects. [9]

Nevertheless, we may infer from Misner, Thorne, and Wheeler's [9] exquisitely detailed geometric treatment of tensors that they did not entirely agree with Lanczos's [13] sentiment. It is also worth mentioning that a beautiful geometric proof of Cauchy's stress theorem (provided by W. Noll in a private communication to the editor) appears in Volume II of Truesdell's *Mechanics of Solids* (formerly Volume VIa/2 of the *Encyclopedia of Physics*) [14] alongside a more traditional algebraic proof thereof. The present authors, too, are of the opinion that there is nothing to be lost in understanding a topic from multiple perspectives (after all, this is one of the pillars of modern pedagogy).

On the topic of pedagogy, both approaches present challenges to the novice, and it seems that the polarization of the community has only exacerbated this confusion [10]. As observed by Sanders [3], it appears to be common practice in the United States to introduce vectors from the geometric approach (as quantities with both magnitude and direction) in high school and lower-division university courses [15–17], and then to switch abruptly to the transformation rule (1) in upper-level university courses where higher-rank tensors appear [6–8]. Some authors may even switch between the two approaches in the same text. For example, Taylor [7] defines tensors by their transformation rule in Section 15.17 of his celebrated *Classical Mechanics*, but he later includes Noll's geometric proof of Cauchy's stress theorem [14] in Section 16.7, evidently deeming the geometric proof superior to the algebraic. Sanders [3] argues that it is better to be consistent, advocating for the geometric approach as definition and the transformation rule as corollary, while conceding that some may prefer the more concrete algebraic approach.

Regardless of which approach one takes, the problem remains that the physical meaning of the transformation rule (1) is difficult for the novice to grasp. For tensors of the second rank (particularly stress and strain), Mohr's circle is commonly used to visualize the transformation rule [18, 19]. While Mohr's circle is a useful tool, it only applies to planar rotations about a single axis, and the angle subtended on the circle is not the actual angle of rotation (those being related by a factor of 2). This has led some educators to develop computer programs and mobile apps designed to illustrate coordinate transformations in three dimensions. Notably, Bischof and Edelbauer [2] have created a graphical interface in which the user may specify the components of a vector in a Cartesian coordinate basis, as well as another set of basis vectors (not necessarily orthonormal), and the program outputs the covariant and contravariant components of the vector in the new basis. Unfortunately, similar programs that attempt to illustrate stress and strain transformations are either static (in the sense that the user specifies the initial tensor components and the rotation to be applied, and the program displays the final results without any intermediate history) or not widely available (*e.g.*, [20]). In response, the present authors have developed applications that illustrate three-dimensional tensor transformations dynamically, in real time. This paper documents the development of these applications and serves as their public debut.

Of particular relevance to the present work, we note that Pirker [21] has used virtual reality (VR) to create a virtual "educational physics laboratory" and has compared the efficacy of the VR experience on mobile devices versus in the classroom. The results of Pirker's study [21] indicate that the mobile experience profits from more flexibility and portability, while the room-scale experience profits from a greater degree of interaction, hands-on experience, and immersion into the virtual environment. Seeking the best of both worlds, the present authors have developed two separate applications to illustrate tensor transformations: one designed for VR headsets such as the Oculus Quest 2 [22] and another web-based app designed for traditional computers and mobile devices. We document the VR and web-based apps in Sections 2 and 3, respectively. Both applications were created using the Unity game engine. In each case, the user manually manipulates a given system using either the hand controller (in a VR headset), the cursor (on a traditional computer), or their finger (on a mobile device), and the relevant tensor components update continuously while the transformations are being performed. It is our hope that these tools will assist students at all levels in understanding tensor component transformations, in both in-person and online learning environments.

2 Virtual reality application: *Explore Tensors* (EXTE)

The main focus of the VR application is a student or instructor's direct interaction with rank-two tensor quantities, and the video transmission of this interaction to third parties (such as a classroom or lecture hall). For this purpose, a runtime and development environment for video games (Unity) with the support of Visual Studio Community and Blender was developed as an application for the VR headset Oculus Quest 2 [22]. The chosen VR headset is an all-in-one system, *i.e.*, no external sensors or cameras are needed. In addition, it offers several possibilities for streaming, which supports the presentation of the game world to third parties.

Since motion sickness can occur in such room-scale VR applications, all menu items are located in the so-called *World Space*. This means that all elements are represented as objects in space and not bound to the user's field of view as in comparable applications. In addition, the elements are placed at a comfortable distance from the user, so that all elements remain neatly arranged and do not restrict the view of the user. This ensures that the user moves as little as possible, which in turn counteracts motion sickness.

In the present stage of this application, which was named *Explore Tensors* (EXTE) [23, 24], the users have two tensor quantities to choose from, which are explored in two separate program levels. In the *Inertia Tensor* level, users can explore the changes of the components of the mass moment of inertia tensor by means of geometric manipulation of an object, *i.e.*, by changing its position, orientation and shape. In the *Stress Tensor* level, the users can view the stress field within a simply loaded cantilever beam. Depending on the position of a reference volume element and of the orientation of the sectional plane, the change in the components of the stress tensor can be observed. In addition, the strain tensor and Mohr's circles for a general three-dimensional state of stresses can be displayed.

Both levels require sensitive manipulations with a high degree of precision, and the hand animation and interaction, which is otherwise very popular in VR applications, is not applicable. Instead, this application animates the controllers, as depicted in Figure 1. As can be seen in Figure 1, a pointing laser is animated on these controllers, which enables precise manipulation of orientation and at the same time defines the direction of the *raycaster*, a function for the interaction with the elements. In addition, when pointing to menu items or other elements, a *gaze pointer*, *i.e.*, a kind of mouse pointer, is displayed on the surfaces. In order to further support the user, the orientation of the VR glasses and the position of the controllers are used to provide additional assistance in the form of *tooltips*, as shown in Figure 2.

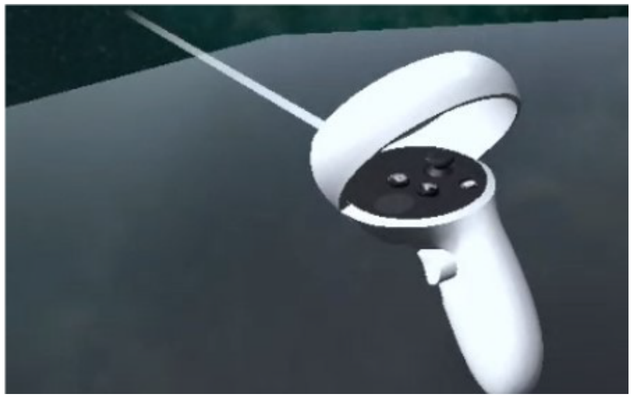

Figure 1. Animation of the right controller with pointing laser.

Figure 2. Tooltips with information on the colors of the coordinate axes (x = red, y = green, z = blue) and interaction buttons.

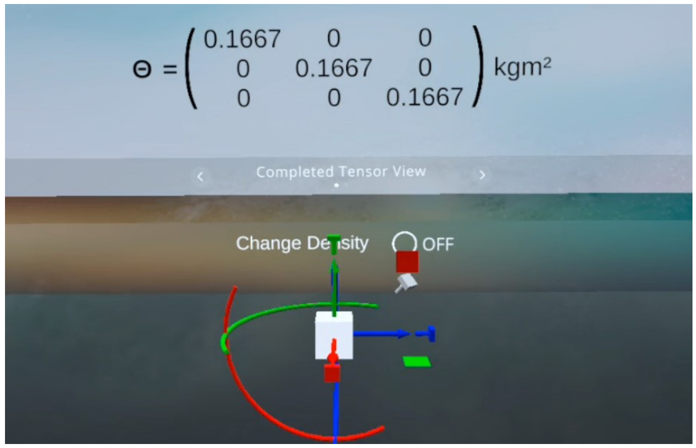

Figure 3. Example object cube with its displayed inertia tensor Θ.

2.1 Program level: Inertia Tensor

The *Inertia Tensor* level is intended to familiarize the user with the inertia tensor in a playful way. By manipulating the position, orientation, and shape of an object, the corresponding change in the tensor components becomes apparent. At the beginning, the user is given the choice between the sample objects cuboid and sphere. The selected object then appears in room-scale, *i.e.*, life-size, as shown in Figure 3 with a cube as the selected object, in front of the user. The matrix representation of the inertia tensor is labeled in the game by the Greek letter Θ.

By means of the so-called *gizmo*, the user can manipulate the sample object. This can be done by pointing to one of the gizmo elements with the pointer and "grabbing" it. The entire object can be moved in all six degrees of freedom provided by the gizmo (see Figure 4; note that Unity uses a left-handed coordinate system). In addition, the object can be scaled as a whole or independently in any direction. As a result, the initial cube can be quickly shaped into a beam, a plate or simply enlarged. The other example object, the sphere, can be transformed into an ellipsoid of any aspect ratio. In Figure 5, the cube has been deformed into a beam by stretching in the z-direction (blue). As a result, the xx- and yy-components increase, and the zz-component increases a little as well due to the volume and corresponding mass increase (the density of the material is uniform).

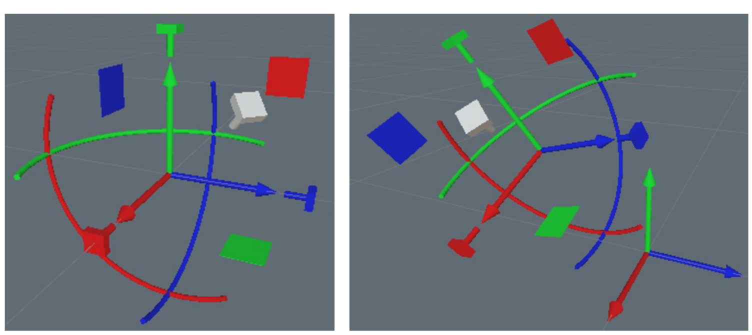

Figure 4. Gizmo in initial position (left) and translated as well as rotated (right). Note that Unity uses a left-handed coordinate system.

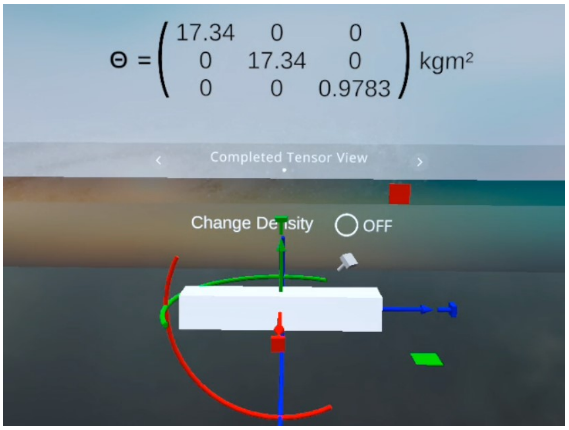

Figure 5. The cube has been transformed into a beam by direction-dependent scaling.

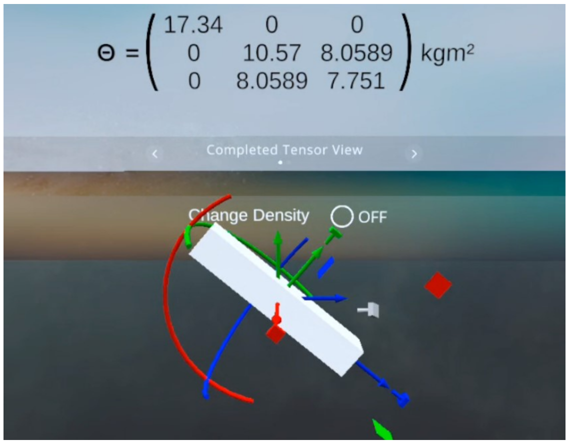

Figure 6. A rotation of the beam changes the components of the inertia tensor.

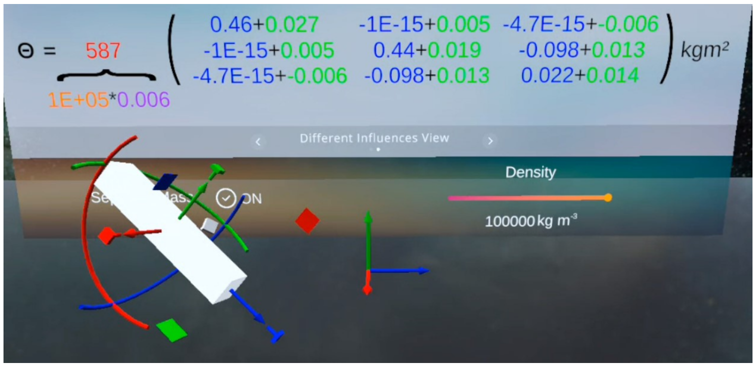

Figure 7. Assignment of transformation-dependent color codes to tensor components.

A rotation around the x-axis changes the components of the inertia tensor and causes products of inertia to develop, as depicted in Figure 6. In Figure 7, the beam has been additionally rotated around the y-axis and moved away from its original position relative to the reference frame. In this way, the user can be shown which changes to the object or the parameters cause which changes in the tensor components. It can be explored, for example, how great the influence of a small volume with high density at some distance to the axis of rotation is, compared to that one of a large volume with low density. For an analysis of the influences of these different operations on the tensor components, they can be displayed in different colors. Changes in the tensor components due to parallel shifts are shown in blue, and those by rotations in green. Other influencing variables are volume (violet), mass (red), and density (orange).

2.2 Program level: Stress Tensor

In this level, the user is given the opportunity to get acquainted with the stress tensor. For this purpose, a simple cantilever beam is provided to the user for virtual experiments. At the beginning, the user can place a point load, represented by an arrow, anywhere on the beam as shown in Figure 8. The parameter menu can be used to adjust its magnitude and the angles at which the force acts.

Due to the resulting bending moment a stress field is induced in the beam. The user is given the opportunity to look into the body with a movable sectional plane and to view the local stress state at any point, represented by an infinitesimal cube. This sectional plane can be manipulated by means of the gizmo. It can be moved freely in the entire volume of the beam and inclined at any desired angle (see Figures 9 and 10). The center of the gizmo represents the point at which the reference volume, the infinitesimal cube, is located and the stress state is determined. In Figure 9 the gizmo is placed exactly in the neutral surface of the symmetric beam, which is why only shear stresses occur. On the left side, the user interface displays Mohr's circles for this state of stress.

The cube that represents the local stress state is highlighted and displayed three-dimensionally in the orientation of the sectional plane for better comprehensibility, as shown in Figures 9 and 10. When tensor components become zero, the associated internal traction vector components also disappear from the corresponding surfaces of the cube. The cube can be rotated on the spot according to the orientation of the gizmo and the local stress state is displayed in the matrix representation of the stress tensor, as depicted in Figure 10. In this figure, the tensor components

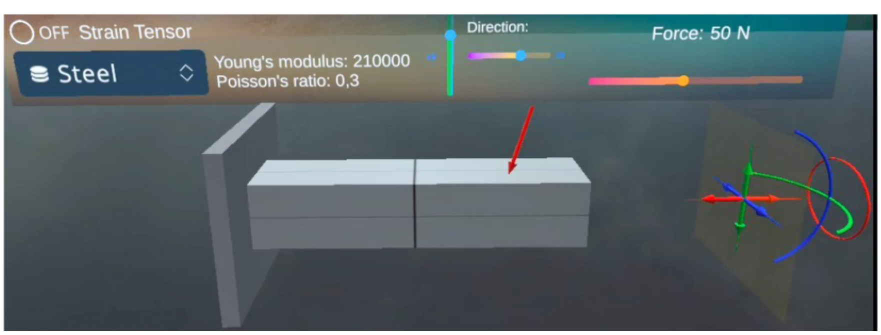

Figure 8. Simple cantilever beam with applied point load represented by the red arrow.

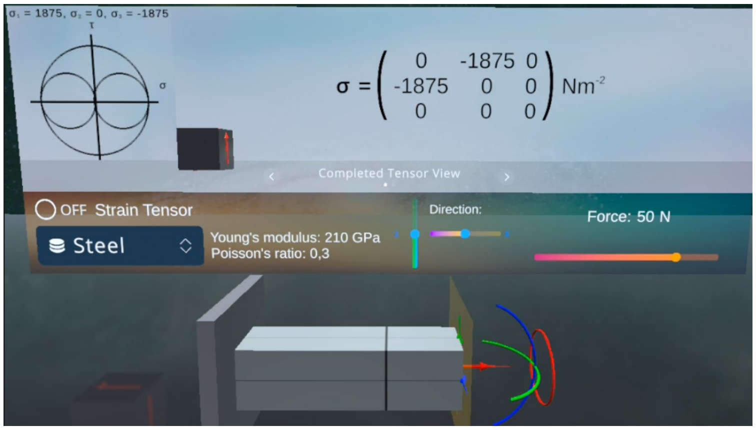

Figure 9. Sectional plane perpendicular to the neutral line of the beam.

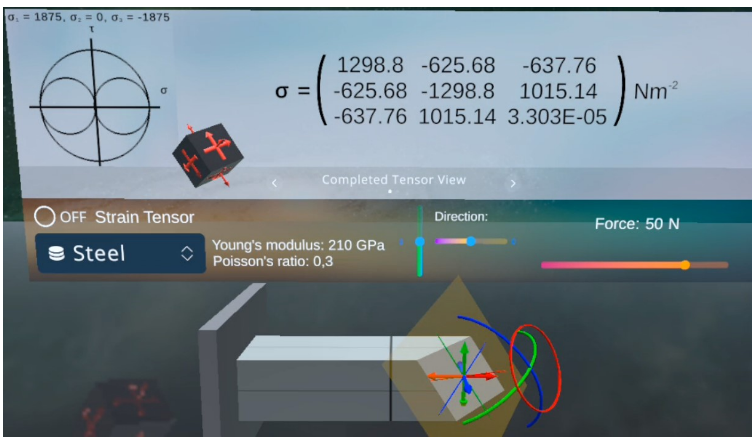

Figure 10. Sectional plane inclined according to the orientation of the gizmo.

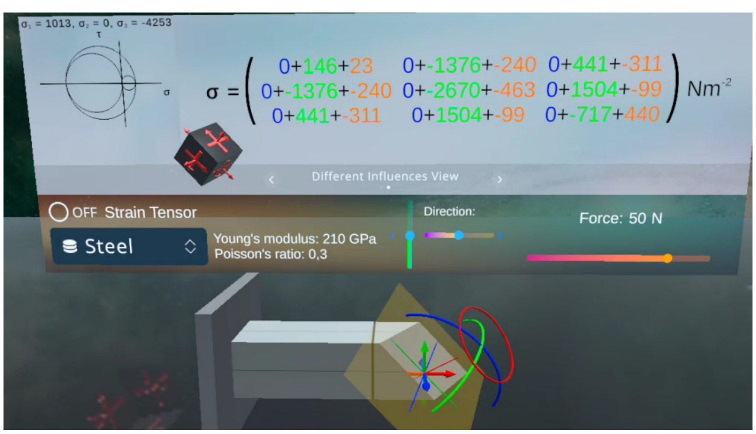

Figure 11. Display mode for the analysis of tensor components.

have changed, but as can be seen from Mohr's circles, the local stress state has not changed at all.

Similar to the *Inertia Tensor* level, the representation can be extended so that the individual influences on the tensor components can be represented by means of color codes, such as which stresses cause bending or torsion at a certain point (see Figure 11). In addition, the toggle *Strain Tensor* on the user interface allows the strain tensor to be displayed for the material properties associated with the selected linear elastic material.

The software for *Explore Tensors* (EXTE) [23, 24] is available online completely free of charge for anyone to use. Interested readers may find the link in the Bibliography under Reference [23]. The only cost associated with *Explore Tensors* (EXTE) [23, 24] is the VR headset. At the time of this writing, the cost of an Oculus Quest 2 [22] starts at $299.00 (U.S. Dollars), although of course this price will vary as time passes and demands change. As alluded to above, it is by no means necessary for each student in an in-person classroom setting to have their own headset. A small number of headsets (or even a single headset) may be shared among several students. Alternatively, the instructor may cast their gameplay experience to a screen for students to view, although this would certainly be less engaging for the students. Ultimately, the cost and economic feasibility of the VR experience will be dictated by the individual institution and instructor's budgetary restrictions.

3 Web-based application: *Tensor transform visualization*

For many learning communities across the globe, the COVID-19 pandemic necessitated (at least for a time) a transition to 100% remote instruction, and this has generated increased interest in pedagogically effective remote teaching strategies. In a completely remote learning environment, many students may not have access to a VR headset—especially students coming from underprivileged backgrounds. For such cases, we have also developed a web-based application similar in spirit to *Explore Tensors* (EXTE) [23, 24] that students can access via the internet on their personal computers or mobile devices. It is tacitly assumed here that most students nowadays have access to the internet, although of course this may not be a valid assumption in some communities.

The web-based application, named *Tensor transform visualization* [25], is markedly simpler than the VR application. The user interface consists of a single cube (as shown in Figure 12) meant to represent an infinitesimal material element subjected to an imagined state of stress, as defined by nine Cartesian components

$$[\boldsymbol{\sigma}_0] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \tag{2}$$

in the initial configuration, in some imagined units of stress. The user may rotate the cube about its centroid using the mouse cursor (on a traditional computer) or their finger (on a mobile device), and the stress components update continuously while the rotation is being performed. Figure 12 shows the element in two different configurations. Each column matrix $\mathtt{e}'_i$ represents the components of one of the rotated basis vectors, while the square matrix $\mathtt{S}$ represents the nine components of the stress tensor in the rotated basis. For example, with the rotated basis shown,

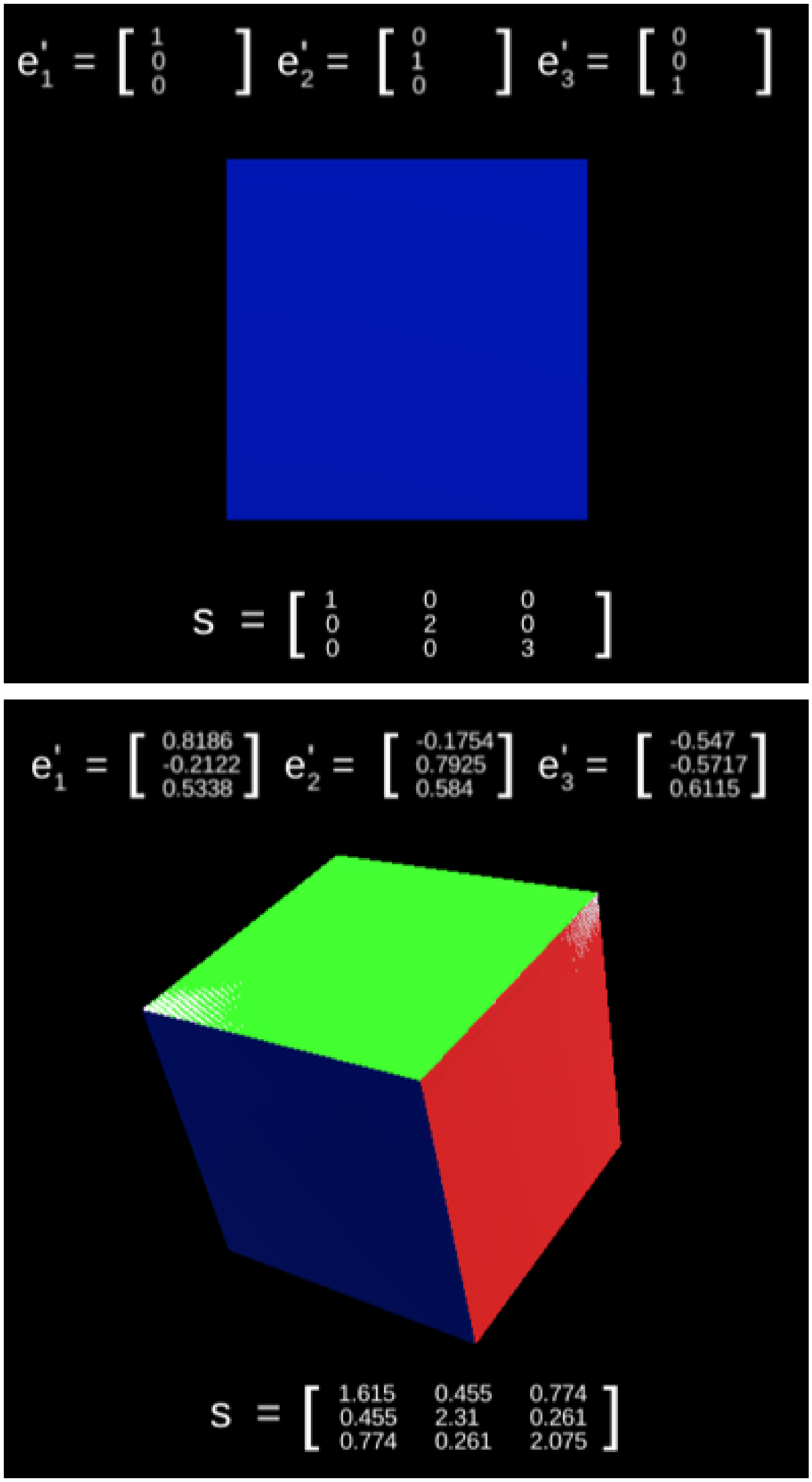

Figure 12. The infinitesimal material element within the web-based application, in two configurations. Each column matrix represents the components of a rotated basis vector. The square matrix represents the nine components of the stress tensor in some imagined units of stress.

we have for the rotation matrix

$$[\mathbf{R}] = \begin{bmatrix} 0.8186 & -0.2122 & 0.5338 \\ -0.1754 & 0.7925 & 0.584 \\ -0.547 & -0.5717 & 0.6115 \end{bmatrix}, \tag{3}$$

and for the transformed stress components

$$[\boldsymbol{\sigma}] = \begin{bmatrix} 1.615 & 0.455 & 0.774 \\ 0.445 & 2.31 & 0.261 \\ 0.774 & 0.261 & 2.075 \end{bmatrix}. \tag{4}$$

It is straightforward to check that $[\boldsymbol{\sigma}] = [\mathbf{R}][\boldsymbol{\sigma}_0][\mathbf{R}]^{\mathrm{T}}$, in accordance with (1).

As of the time of this writing, the traction vectors are not illustrated on the faces of the material element in the web-based application, although that modification is currently in development. For now, the faces are color-coded in the same manner as *Explore Tensors* (EXTE) (x = red, y = green, z = blue), with the exception that the displayed tensor components have been modified to make this a right-handed coordinate system. Another planned modification is the functionality for the user to specify the initial stress components.

The web-based application *Tensor transform visualization* [25] is available online completely free of charge for anyone to use. Interested readers may find the link in the Bibliography under Reference [25].

4 Summary and conclusion

This paper presents the public debut of two pedagogical tools the authors have developed to illustrate three-dimensional, rank-two tensor component transformations in real time. The first, *Explore Tensors* (EXTE) [23, 24], is a virtual reality application designed for VR headsets such as the Oculus Quest 2 [22]. The second, *Tensor transform visualization* [25], is a simpler web-based application designed to be accessed via the internet on traditional computers or mobile devices. It is hoped that these tools will be used to assist students at all levels in understanding tensor component transformations, in both in-person and online learning environments. A rigorous assessment of the educational effectiveness of these tools is planned for future work.

Authorship declaration

Credit for the initial conceptualization of this work is shared equally by Günter Bischof and John Sanders. The VR application *Explore Tensors* (EXTE) [23] was developed by Markus Wieser as part of his Bachelor's thesis [24] under the supervision of Günter Bischof at the University of Applied Sciences FH JOANNEUM (Austria). The web-based application *Tensor transform visualization* [25] was developed by Serop Kelkelian as part of a semester-long independent study course under the supervision of John Sanders at California State University, Fullerton (United States).

Bibliography

[1] Grossfield, A., 2000. "Mathematical definitions: What is this thing?". In Proceedings of the 107th American Society for Engineering Education Annual Conference and Exposition (ASEE 2000).

[2] Bischof, G., and Edelbauer, B., 2018. "Coordinate transforms and dual bases: a teaching aid for undergraduate engineering students". In Proceedings of the 125th American Society for Engineering Education Annual Conference and Exposition (ASEE 2018).

[3] Sanders, J., 2018. "Demystifying tensors: A friendly approach for students of all disciplines". In Proceedings of the 125th American Society for Engineering Education Annual Conference and Exposition (ASEE 2018).

[4] Jeffreys, H., 1931. *Cartesian Tensors*. Cambridge University Press.

[5] Gurney, C., 1984. "Representation of tensors by matrices". *International Journal of Mathematical Education in Science and Technology,* ***15***(6), pp. 703–711.

[6] Griffiths, D. J., 1999. *Introduction to Electrodynamics*. Prentice Hall.

[7] Taylor, J. R., 2005. *Classical Mechanics*. University Science Books.

[8] Sadd, M. H., 2009. *Elasticity: Theory, Applications, and Numerics*. Academic Press.

[9] Misner, C. W., Thorne, K. S., and Wheeler, J. A., 1970. *Gravitation*. W. H. Freeman and Company.

[10] Jeevanjee, N., 2010. *An Introduction to Tensors and Group Theory for Physicists*. Springer.

[11] Chadwick, P., 1976. *Continuum Mechanics: Concise Theory and Problems*. Dover Publications.

[12] Norton, J. D., 1993. "General covariance and the foundations of general relativity: eight decades of dispute". *Reports on Progress in Physics,* ***56***, pp. 791–858.

[13] Lanczos, C., 1949. *The Variational Principles of Mechanics*. Dover Publications.

[14] Truesdell, C., ed., 1984. *Mechanics of Solids, Volume II: Linear Theories of Elasticity and Thermoelasticity, Linear and Nonlinear Theories of Rods, Plates, and Shells*. Springer-Verlag.

[15] Giancoli, D. C., 1998. *Physics: Principles with Applications*. Prentice Hall.

[16] Resnick, R., Halliday, D., and Krane, K. S., 2002. *Physics*. John Wiley and Sons.

[17] Hibbeler, R. C., 2007. *Engineering Mechanics: Dynamics*. Pearson.

[18] Gere, J. M., and Goodno, B. J., 2012. *Mechanics of Materials*. Cengage Learning.

[19] Beer, F. P., Johnston, E. R., Dewolf, J. T., and Mazurek, D. F., 2012. *Mechanics of Materials*. McGraw Hill.

[20] Hennessey, M., and Hacker, L., 2006. "Dynamic 3 D visualization of stress tensors". In Proceedings of the 113th American Society for Engineering Education Annual Conference and Exposition (ASEE 2006).

[21] Pirker, J., and Lesjak, I., 2017. "An educational physics laboratory in mobile versus room scale virtual reality–a comparative study". *International Journal of Online Engineering,* ***13***, p. 106.

[22] Oculus Quest 2 by Meta. Accessed Jan 05, 2022. https://www.oculus.com/quest-2/.

[23] Bischof, G., and Wieser, M. Explore Tensors (EXTE). Available online via OneDrive, https://bit.ly/Explore-Tensors-EXTE-US.

[24] Wieser, M., 2021. Visualisierung der Auswirkungen von Koordinatentransformationen auf Tensorkomponenten mittels Virtueller Realität (Making use of virtual reality to illustrate the effects of coordinate frame rotations on tensor components). Bachelor's thesis, University of Applied Sciences FH JOANNEUM.

[25] Sanders, J., and Kelkelian, S. Tensor transform visualization. Available online via GitHub, https://jdubulew.github.io/tensor-transform-visualization/.