Comprehensive Journal of Humanities and Educational Studies
المجلة الشاملة للدراسات الإنسانية التربوية
Volume 1, Issue 1, 2025, Pages: 19-27
Journal homepage :http://cjhes.com.ly/index.php/cjhes/en/index

Matrix Transformations and Their Geometric Effects in 2D Space

Zaynab Ahmed Khalleefah *
Faculty of Sciences, University of Gharyan, Libya

تحويلات المصفوفات وتأثيراتها الهندسية في الفضاء ثنائي الأبعاد

زينب أحمد خليفة *
كلية العلوم الأصابعة، جامعة غريان، ليبيا

*Corresponding author: zzainabah87@gmail.com

Received: April 22, 2025	Accepted: July 02, 2025	Published: July 10, 2025

Abstract:
This paper explores linear transformations in two-dimensional space, focusing on how 2×2 matrices act on vectors and shapes. We review definitions of vectors and matrices, then examine common transformations: scaling, rotation, reflection, shear, and projection. Each type's matrix form and effect on points are given, with example equations. We show how basic shapes (square, triangle, circle) change under transforms, and discuss eigenvalues and invariant directions. Composition of transforms (matrix multiplication) is explained, noting that the order matters. Applications in computer graphics, physics, and robotics are highlighted. Figures and tables summarize key transformations, and practice problems are provided. Overall, we emphasize visualization and intuitive understanding of how matrices warp the plane.

Keywords: Matrix transformation, linear algebra, 2D geometry, scaling, rotation, shear, projection, eigenvalues, computer graphics.

ملخص:
تستكشف هذه الورقة البحثية التحويلات الخطية في الفضاء ثنائي الأبعاد، مع التركيز على كيفية تأثير مصفوفات 2×2 على المتجهات والأشكال. نستعرض تعريفات المتجهات والمصفوفات، ثم ندرس التحويلات الشائعة: القياس، والدوران، والانعكاس، والقص، والإسقاط. يُعطى شكل كل نوع من أنواع المصفوفات وتأثيره على النقاط، مع أمثلة على المعادلات. نوضح كيف تتغير الأشكال الأساسية (المربع، المثلث، الدائرة) تحت تأثير التحويلات، ونناقش القيم الذاتية والاتجاهات الثابتة. نشرح تركيب التحويلات (ضرب المصفوفات)، مع مراعاة أهمية الترتيب. كما نسلط الضوء على التطبيقات في رسومات الحاسوب، والفيزياء، والروبوتات. تلخص الأشكال والجداول أهم التحويلات، ونقدم مسائل تدريبية. بشكل عام، نركز على التصور والفهم البديهي لكيفية تشويه المصفوفات للمستوى.

الكلمات المفتاحية: تحويل المصفوفات، الجبر الخطي، الهندسة ثنائية الأبعاد، القياس، الدوران، القص، الإسقاط، القيم الذاتية، رسومات الحاسوب.

Introduction

Linear algebra in two dimensions studies vectors in $\mathbb{R}^2$ and how matrices map them. A 2×2 matrix A acts on a vector x by producing Ax, a new vector. In this view, a matrix represents a function $f(x) = Ax$ from $\mathbb{R}^2$ to itself. Margalit and Rabinoff (2019) explain that "A can be thought of as a function with independent variable x and dependent variable $b = Ax$". In practice, many geometric operations are linear transformations. For example, a matrix can model scaling, rotation, reflection, or shear of the plane. These transformations are fundamental in fields like computer graphics and robotics. For instance, rotation matrices are used "extensively for computations in … physics and computer graphics". Computer graphics often applies translation, rotation, and scaling to images, while robotics uses matrices to represent positions and orientationsmotion.cs.illinois.edu. This paper surveys the math behind these 2D transformations, illustrates their geometric effects, and discusses applications.

Mathematical Background

A 2D vector is an ordered pair (x, y) that can be written as a column $\binom{x}{y}$. A 2×2 matrix has form

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

with four real entries. The matrix acts on a vector by the rule

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} ax + by \\ cx + dy \end{pmatrix}$$

This is matrix-vector multiplication. It combines the matrix rows with the vector's components (the row–column rule) to yield a new 2D vector. For example, if $A = \begin{pmatrix} 2 & 1 \\ 0 & 1 \end{pmatrix}$ and $v = (x, y)$, then $Av = (2x + y, 0x + 1y) = (2x + y, y)$. In linear algebra, such a map $v \mapsto Av$ is called a linear transformation. It is a function that preserves vector addition and scalar multiplication. Concretely, $T(u + v) = T(u) + T(v)$ *and* $T(c\,v) = c\,T(v)$ for all vectors u, v and scalars c.

Basic matrix operations include addition and multiplication. Matrix multiplication is not commutative, meaning $AB \neq BA$ in general. The product of two matrices A and B corresponds to doing their transformations in sequence. If T has matrix A and U has matrix B, then the composition $T \circ U$ has matrix AB. Equations are often written as $b = Ax$ *or* $y = Ax$, showing how x is mapped to Ax. For example, if $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ and $x = \begin{pmatrix} x \\ y \end{pmatrix}$ then

$$Ax = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} ax + by \\ cx + dy \end{pmatrix}$$

This linear action can be interpreted geometrically as moving and stretching points in the plane (Fig. 1).

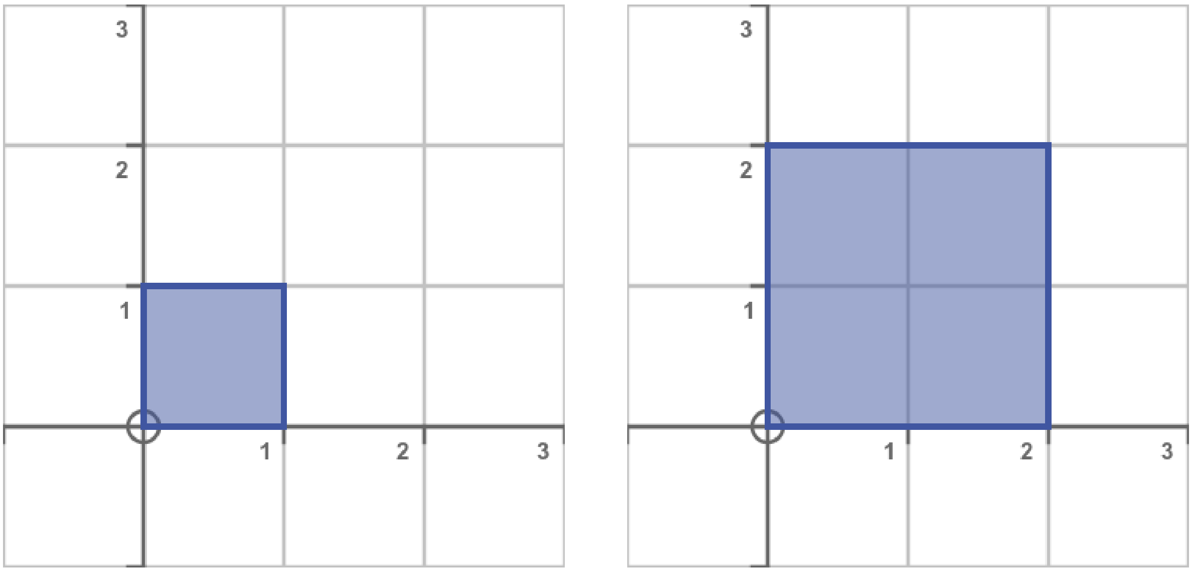


Figure 1 Geometric interpretation of matrix-vector multiplication. (A 2D vector is mapped to a new vector via the matrix.) Source: GraphicMaths (visualization of Ax).

Types of Matrix Transformations in 2D

Linear transformations in the plane come in standard types. Here we describe several and give their matrix forms and effects.

Scaling: This stretches or compresses distances along the axes. A scaling matrix is diagonal:

$$S = \begin{pmatrix} sx & 0 \\ 0 & sx \end{pmatrix}$$

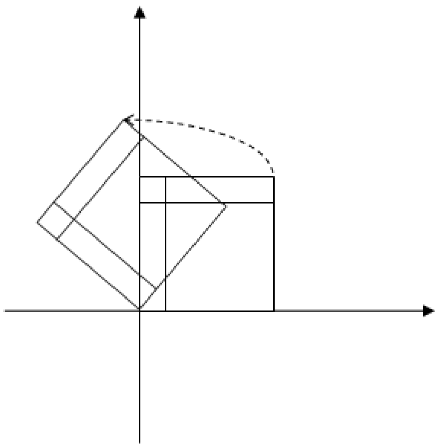
If $sx > 1$, the *x*-direction is stretched; if $0 < s_x < 1$, it is compressed. Similarly s_y scales the *y*-direction. For example,

$$S = \begin{pmatrix} 2 & 0 \\ 0 & 3 \end{pmatrix}$$

doubles *x*-coordinates and triples *y*-coordinates of any point. Under such scaling, a unit square becomes a larger (or smaller) rectangle (Fig. 2). GraphicMaths notes that matrices can describe "scaling, rotating, [and] skewing" of objects.

Rotation: A rotation turns all points by an angle θ about the origin. The rotation matrix is

$$\mathrm{R}(\theta) = \begin{pmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{pmatrix}$$



Multiplying a vector by *R(θ)* rotates it counterclockwise by θ (keeping the origin fixed). For instance, $\mathrm{R}(90°) \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} -y \\ x \end{pmatrix}$. Rotation preserves lengths and orientation. In fact, "rotation matrices have determinant +1". Figure shows a unit square rotated by 10°.

Reflection: A reflection flips the plane across a line through the origin. The matrix for reflecting across a line at angle $\theta/2$ is

$$\mathrm{M} = \begin{pmatrix} \cos(2\theta) & \sin(2\theta) \\ \sin(2\theta) & -\cos(2\theta) \end{pmatrix}$$

For example, reflecting across the *x*-axis (horizontal) is $M = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$. Reflections have determinant -1 and reverse orientation. Under reflection, lengths are preserved but the figure is "mirrored".

Shearing: A shear transformation slides the plane in one direction, turning rectangles into parallelograms. In 2D there are horizontal or vertical shears. A horizontal shear by factor *k* uses

$$H = \begin{pmatrix} 1 & k \\ 0 & 1 \end{pmatrix}$$

so $(x, y) \mapsto (x + ky, y)$. A vertical shear by k uses $\begin{pmatrix} 1 & 0 \\ k & 1 \end{pmatrix}$. A shear distorts shapes (e.g. squares become parallelograms) but preserves area. In a shear, all horizontal lines remain parallel and equally spaced, so the grid lines stay evenly spaced. (Shear is an affine transform; without translation, it is still linear in matrix form).

Orthogonal Projection: A projection sends every point to a fixed line or axis (through the origin) and "flattens" the other direction. For example, projecting onto the *x*-axis is

$$P_x = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

sending $(x,y) \to (x,0)$. A projection matrix satisfies $P^2 = P$ (idempotent). Its eigenvalues are 1 (along the line of projection) and 0 (perpendicular to it). Thus points on the axis stay fixed while others collapse onto it. Orthogonal projections preserve length *along the line* and send perpendicular components to zero.

Table 1 Common 2D linear transformations and their matrices/effects

Transformation	Matrix A	Geometric Effect
Scaling	$\begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix}$	Stretch by s_x in x, s_y in y
Rotation by θ	$\begin{pmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{pmatrix}$	Turn by θ about origin
Reflection across line at angle $\frac{\theta}{2}$	$\begin{pmatrix} \cos(2\theta) & \sin(2\theta) \\ \sin(2\theta) & -\cos(2\theta) \end{pmatrix}$	Mirror across that line
Shear (horizontal, factor k)	$\begin{pmatrix} 1 & k \\ 0 & 1 \end{pmatrix}$	Slant shape horizontally, preserve area
Orthogonal projection onto axis/line	$P^2 = P$	Collapse points onto a line (flatten)

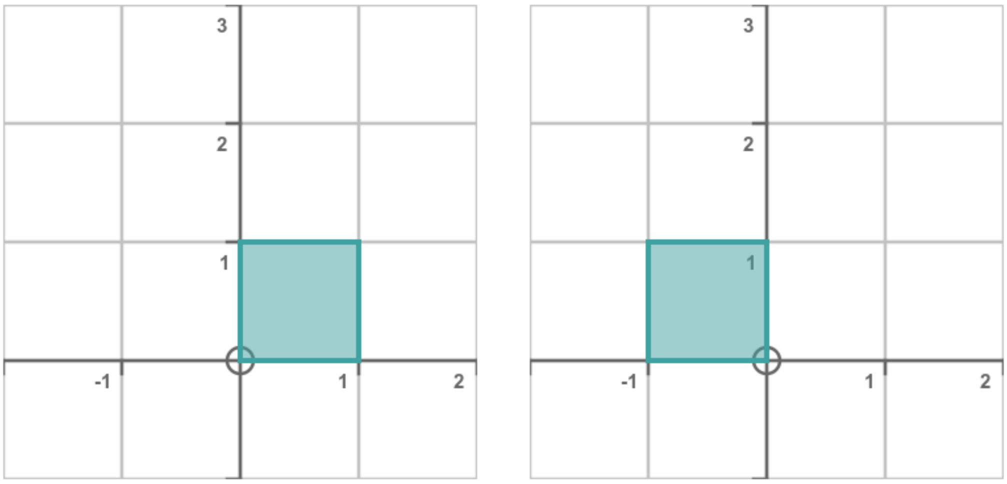


Figure 2 Unit square scaled by 2 in x and 3 in y (blue to red). Source: GraphicMaths (scaling matrix example).

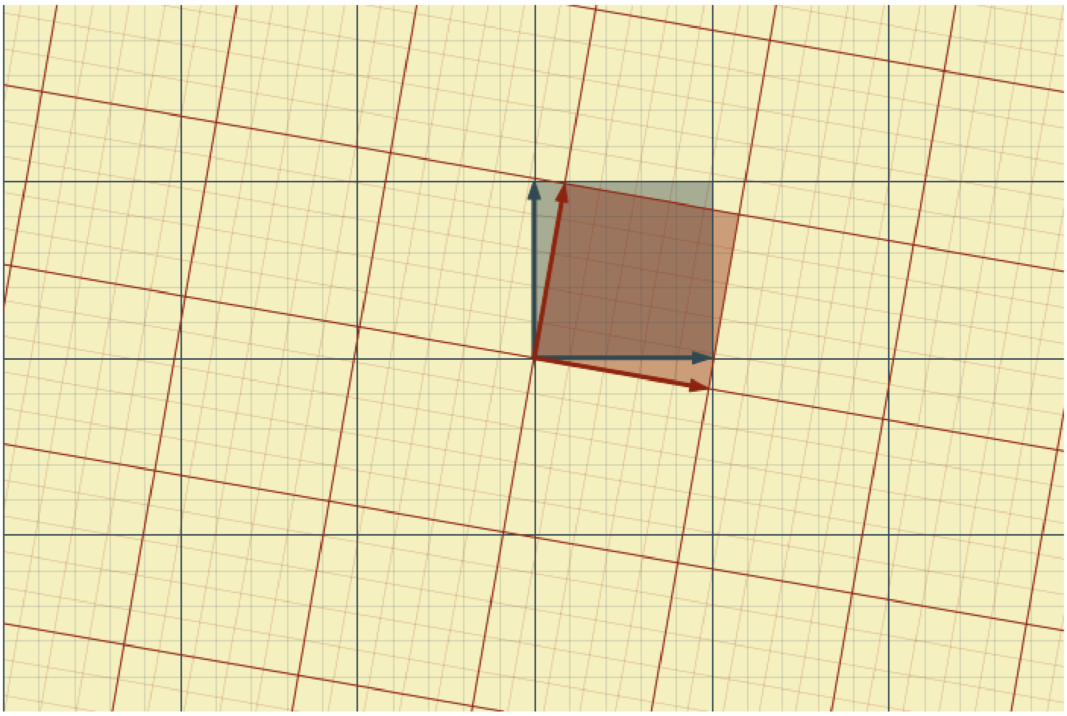


Figure 3 Unit square rotated by 10° (blue to red). Source: SciPython.

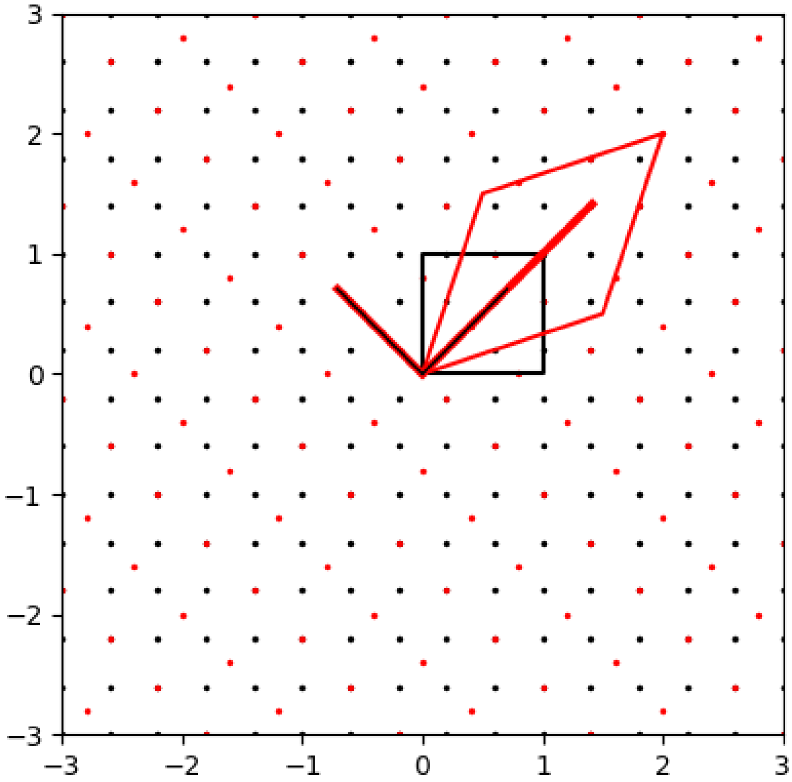


Figure 4 Unit square sheared horizontally (k=0.5). Source: SciPython.

Transformation of Basic Shapes

Applying a matrix to all points of a shape deforms the shape accordingly. We consider a few simple shapes:

- **Unit Square:** The points $(0,0), (1,0), (1,1), (0,1)$ form a square. Under a linear transform A, these map to $\{A(0,0), A(1,0), A(1,1), A(0,1)\}$. Since $A(0,0) = (0,0)$, the origin stays fixed. The other corners move; e.g. with a shear $A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$, the top side becomes slanted. Figure 5 shows a square first rotated and then sheared. After rotation by 10°, the square becomes a rotated parallelogram; applying an additional shear slants it further. (Problems: compute the images of the corners under a given matrix.)
- **Triangle:** A triangle's vertices $\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$ are each transformed by A. For example, if $A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$ (a 90° rotation), then $(x, y) \mapsto (-y, x)$. The new triangle is just the original one rotated or scaled. One can apply any of the matrices above to a triangle and plot the result.
- **Circle (Regular Polygon Approximation):** A circle can be approximated by many points on its circumference (or a regular polygon). Under a linear map, these points move to an ellipse-like shape in general. For instance, a circle of radius 1 centered at origin becomes an ellipse if A has unequal eigenvalues. If the map is isotropic (uniform scaling or rotation), the image remains a circle.

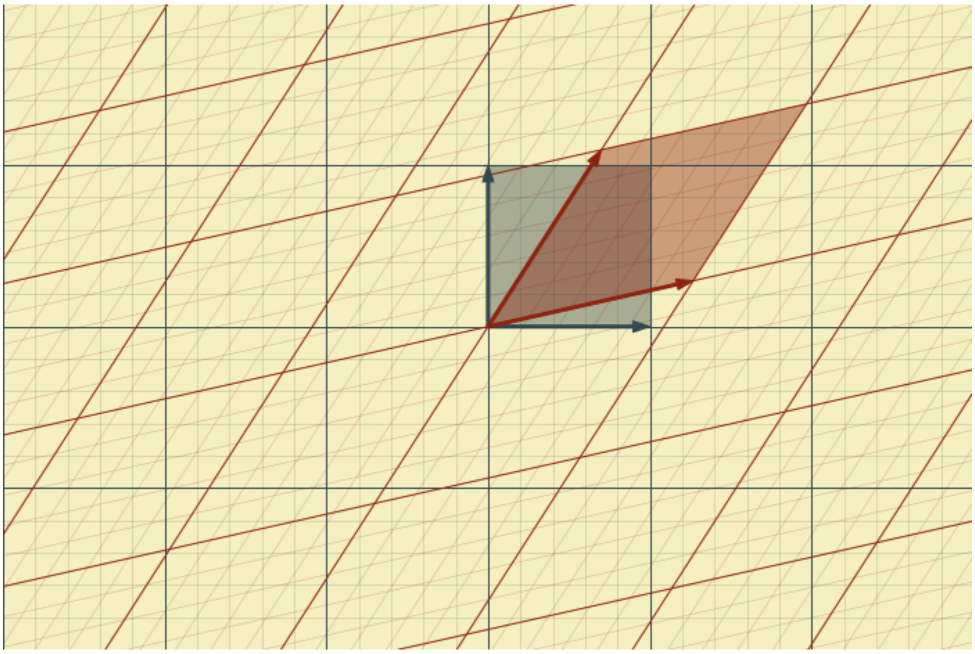


Figure 5 Unit square under a 10° rotation (blue) followed by a horizontal shear (red). Source: SciPython.

Eigenvalues, Eigenvectors, and Invariant Lines

An eigenvector of a matrix A is a nonzero vector v whose direction remains unchanged by A. Formally $Av = \lambda v$, where λ is the corresponding eigenvalue. Geometrically, v is only stretched or shrunk (and possibly reversed) by the factor λ. For example, in a shear transformation only the horizontal axis may be fixed; any vector along that axis is an eigenvector of eigenvalue 1. Wikipedia explains that eigenvectors "have their direction unchanged (or reversed) by a given linear transformation". If $\lambda > 0$, the eigenvector's direction stays the same; if $\lambda < 0$, it is flipped 180°.

To find eigenvalues for a 2×2 matrix $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, we solve the characteristic equation $\det(A - \lambda I) = 0$. This leads to the quadratic $\lambda^2 - (a + d)\lambda + (ad - bc) = 0$. Solving yields two (possibly equal) eigenvalues λ_1, λ_2. Then solve $(A - \lambda_i I)v = 0$ for each to get the eigenvectors.

Eigenvectors have important geometric meaning: they point along *invariant lines* of the transformation. Points on an eigenvector line are only scaled. For example, a reflection across a line has eigenvalues 1 (along the line of reflection) and -1 (perpendicular to it). A projection matrix has eigenvalues 1 (for directions it keeps) and 0 (for collapsed directions). Table 2 gives some examples of simple matrices with their eigenvalues and geometric interpretations.

Table 2 Examples of eigenvalues/eigenvectors for specific 2×2 transformations and their geometric meaning. (Eigenvalues from standard formulas.)

Matrix *A*	**Eigenvalues**	**Eigenvectors (Direction)**	**Meaning**
$\begin{pmatrix} k & 0 \\ 0 & k \end{pmatrix} (scale)$	k,k	Any direction	Uniform scale by *k*.
$\begin{pmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{pmatrix} (rotation)$	Complex if $\theta \neq 0, \pi$	none real (except 0 vector)	Pure rotation, no real invariant line.
$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} (reflect\ across\ y = x)$	1, -1	Directions long $y = x$ and $y = -x$	Reflects points across the line $y = x$.
$\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} (shear)$	1,1	Horizontal axis only	Shear fixes horizontal direction (λ=1)
$\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} (proj.\ to\ x{-}axis)$	1,0	*x*-axis & vertical axis	

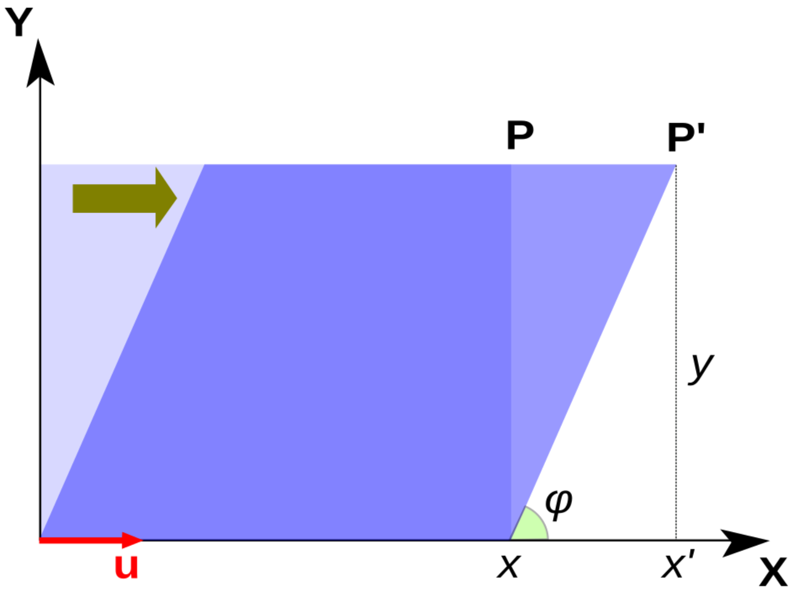


Figure 6 Eigendirections (red) of a shear transformation (black grid). The horizontal vector is an eigenvector. Source: Wikimedia Commons (image by B. Vanderbeke).

Composition of Transformations

Applying two transformations in succession corresponds to multiplying their matrices. If A and B are 2×2 matrices for transformations T and U, then $T(U(x)) = (AB)x$. In fact, the standard matrix for the composition $T \circ U$ is AB. Thus one can combine effects by matrix multiplication. However, matrix multiplication is generally not commutative: $AB \neq BA$. This means the order of transformations matters. For example, rotating then scaling gives a different result than scaling then rotating (unless the matrices commute). Associativity does hold, so $A(BC) = (AB)C$.

Applications in Real Life and Computer Graphics

Matrix transformations are widely used in practice. In computer graphics, every 2D or 3D scene uses linear transforms. Objects, cameras, and lights are positioned by translating, rotating, and scaling their coordinates. As the CS Field Guide notes, "the most common [graphics transforms] are translation, rotation (spinning it) and scaling (changing its size)". Rather than moving each point individually, graphics systems multiply all coordinate vectors by a transformation matrix. Rotation matrices in particular "are used extensively for … computations in … computer graphics". For example, sprite images in a video game can be rotated and scaled smoothly by 2×2 matrices (Fig. 11).

In physics, 2D matrices model rigid-body motion in a plane. A rotation matrix can represent the orientation of a spinning object. Projections are used in optics and graphics to project a 3D scene onto a 2D plane (orthographic or perspective projections use 3×3 matrices for 2D screens). In control systems and robotics, coordinate frames are transformed using matrices. According to one robotics text, "vectors and matrices are used in robotics to represent 2D and 3D positions, directions, rigid body motion, and coordinate transformations". For instance, converting from a robot's joint angles to its hand's world coordinates involves multiplying by rotation/translation matrices.

Many other fields use linear transforms. In data analysis, projections reduce dimensionality (PCA involves eigen-decomposition). In image processing, convolutions and filters are linear operations.

Table 3 Real-world uses of 2D transformations. (Based on applications in graphics, robotics, physics.)

Transform Type	Examples of Real-World Use
Scaling	Zooming images, resizing maps, adjusting font size (UI).
Rotation	Rotating sprites in games, compass direction, coordinate frames in physics.
Reflection	Flipping images (mirror effect), encryption algorithms, symmetry analysis.
Shear	Slanting fonts (italic style), simple image distortion effects.
Projection	3D graphics projection to screen, shadow casting (drop shadows).

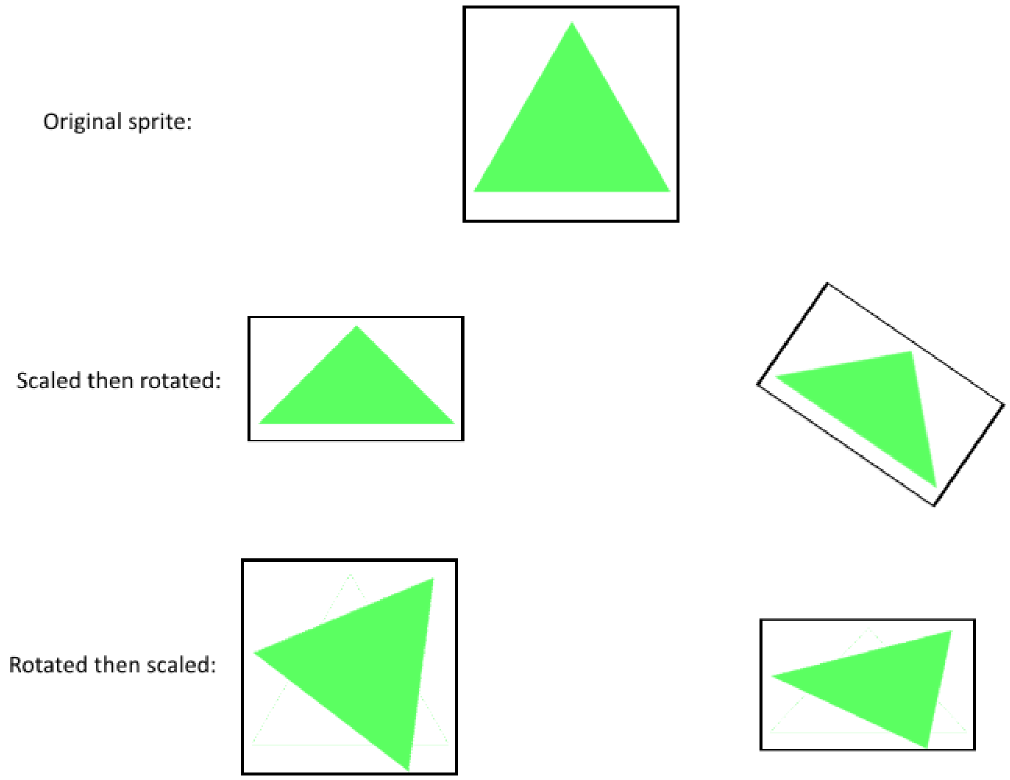


Figure 7 Example of a 2D sprite (image grid) undergoing rotation and scaling.

Discussion

We have seen that 2×2 matrices provide a powerful way to transform the plane. By visualizing their action on simple shapes and grids, one gains intuition about each transformation's effect. Figures and grids make these effects clear, e.g. showing how a square becomes a parallelogram under shear. Visualization tools (e.g. Python with Matplotlib, GeoGebra) can help students explore these ideas interactively.

A limitation is that pure 2×2 matrices only include linear transformations through the origin. Translations (shifts) require using homogeneous coordinates and 3×3 matrices or affine transforms. Also, non-linear warping (like perspective distortion) cannot be done with a single linear matrix. In an educational setting, however, focusing on linear cases is valuable: it illustrates core concepts like eigenvectors and composition simply.

This paper treated several fundamental cases. In practice, one often combines them. We saw that the order of multiplication matters, which is crucial in graphics pipelines. Understanding invariant lines via eigenvectors is key to many algorithms (e.g. finding principal axes). Experimenting with these transformations deepens understanding; for instance, one can write code to apply random matrices to a grid and observe the patterns.

Conclusion

Matrix transformations are essential tools in 2D geometry. We covered the main types (scaling, rotation, reflection, shear, projection) and how they change shapes. We emphasized that eigenvalues/eigenvectors reveal invariant directions in these transforms. Visualization of these effects, through figures and examples, makes the concepts clear. Key takeaways include:

- **Matrix as function:** A 2×2 matrix defines a linear map sending input vectors to outputs, which can be visualized geometrically.
- **Types of transforms:** Scaling, rotation, etc., each has a simple matrix form and specific effect (e.g. rotation matrices preserve lengths).

- **Eigenvectors:** Directions that stay fixed (up to scale) under the transform. They reveal "invariant lines" of the action.
- **Composition matters:** Applying B then A corresponds to AB, and $AB \neq BA$ generally.
- **Applications abound:** From graphics and robotics to data science, these operations are everywhere.

Further study: Readers may extend this to three dimensions (3×3 matrices) and affine transformations (translation included). Nonlinear transformations (e.g. perspective, projective) also have rich geometry. Exercises, such as applying specific matrices to shapes or computing eigenvectors by hand, reinforce the concepts.

Appendix

Practice Problems:

1. **Compute image of a triangle.** Given triangle with vertices $(0,0), (1,0), (0,1)$ and matrix $A = \begin{pmatrix} 2 & 1 \\ -1 & 3 \end{pmatrix}$, find the new vertices *A(v)*.
2. **Eigenvalues of a shear.** For $S = \begin{pmatrix} 1 & k \\ 0 & 1 \end{pmatrix}$, show its eigenvalues and explain the invariant direction.
3. **Order of operations.** Let $S = \begin{pmatrix} 1 & k \\ 0 & 1 \end{pmatrix}$ (90° rot) and $D = \begin{pmatrix} 2 & 0 \\ 0 & 0.5 \end{pmatrix}$ (scale). Compute *DR* and *RD*, and interpret the effect on a sample point (1,0).
4. **Projection check.** Verify that $P = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ satisfies $P^2 = P$ and find its eigenvalues and eigenvectors.
5. **Shape transformation.** The unit circle $x^2 + y^2 = 1$ is transformed by $A = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$**.** What is the image of this circle?

Worked Example: Apply $A = \begin{pmatrix} 3 & -2 \\ 1 & 4 \end{pmatrix}$ to the unit square corners. We get

$$A\begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, A\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \end{pmatrix}, A\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 5 \end{pmatrix}, A\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -2 \\ 4 \end{pmatrix}$$

Plotting these shows the transformed parallelogram (see Table 1 for checking shape).

References

1. Hill, C. (2025). *Visualizing a linear transformation in the Cartesian plane*. SciPython. Retrieved from https://scipython.com/books/book2/chapter-6-numpy/examples/visualizing-linear-transformations/
2. Lax, P. D. (2007). *Linear Algebra and Its Applications* (2nd ed.). John Wiley & Sons.
3. Margalit, D., & Rabinoff, J. (2019). *Interactive Linear Algebra*. Available at Georgia Tech. Retrieved from https://textbooks.math.gatech.edu/ila
4. MathWorks. (2024). *2D Transformation matrices*. GraphicMaths. Retrieved from https://www.graphicmaths.com/pure/matrices/matrix-2d-transformations/
5. SciPython (Hill, C.). (2025). *Linear transformations on the two-dimensional plane*. Retrieved from https://scipython.com/blog/linear-transformations-on-the-two-dimensional-plane/
6. *Shear mapping*. (n.d.). In *Wikipedia*, from https://en.wikipedia.org/wiki/Shear_mapping
7. *Rotation matrix*. (n.d.). In *Wikipedia*. from https://en.wikipedia.org/wiki/Rotation_matrix
8. Strang, G. (2016). *Introduction to Linear Algebra* (5th ed.). Wellesley-Cambridge Press.
9. *Transformations in Computer Graphics*. (n.d.). In *CS Field Guide*. Retrieved [Month Day, Year], from https://www.csfieldguide.org.nz/en/chapters/computer-graphics/graphics-transformations/
10. Vanderbeke, B. (2019). *Eigencircle and rotation and scaling* [Image]. Wikimedia Commons. Retrieved from https://commons.wikimedia.org/wiki/File:Eigencircle_and_rotation_and_scaling.png