

An Interactive Graphics Tool to Support Student Learning of Matrix-based 2D Transformation Calculations

Matt Smith

Department of Informatics, School of Informatics and Engineering
Institute of Technology Blanchardstown, Dublin 15, Ireland
matt.smith@itb.ie

Abstract

“Learning by doing” (for example see [MAT-88]) is an informal way to describe a constructivist philosophy to the design of educational tools. This paper presents the current state of a project to develop an interactive graphical tool “LEG” to support computer graphics students learning the use of matrix arithmetic to calculate two-dimensional geometric transformations. The tool supports a simple form of “cognitive scaffolding” [VYG-78] by providing button and argument interaction (scale by 20 degrees clockwise) as well as provide direct editing of a transformation matrix, thus students can “use” the tool to understand, then “investigate and describe” the matrices generated by when different transformation buttons and arguments are entered, then finally “manipulate the tool free of (button-based) assistance” by directly creating an editing matrices and applying them to the animated graphical object.

Categories and Subject Descriptors (according to ACM CCS):

K.3.1 Computer Uses in Education & I.3.5 Computational Geometry and Object Modeling

1 Introduction

1.1 Motivation for this tool

While a basic level of mathematics remains an entry requirement for computing courses, most students entering third level education are neither confident nor highly successful in immediately engaging in the applied mathematics required for two- and three- dimensional computer graphics [TIM-02]. For this reason, which supports the author’s personal experience of teaching introductory computer graphics courses for many years, clearly there is motivation for innovative ways to help students understand and become proficient in the motivation, concepts and mechanics of mathematics for two-dimensional graphics transformations.

The aim of the work described in this paper is simply to help students quickly understand the results, and calculations for the use of three-by-three matrix arithmetic to calculate two-dimensional graphics transformations. The work has been the design and implementation of an interactive learning environment called “LEG: Learning Environment for Graphics”.

1.2 Related educational research

The constructivist educational philosophy (for example see [MAT-88]) emphasises the need for students to construct their own understanding through experience and experiment. Computer-based learning systems based on the constructivist philosophy have been developed for a range of subject domains, including guided discovery learning of music analysis [HOL-89] and exploratory learning environments for school physics [LST-99]. From such a background the work described in this paper has a goal of providing an interactive learning environment which students can explore and manipulate to discover for themselves the relationship between patterns of changes in three-by-three matrices and their application to graphical objects to affect geometric transformations.

Vygotsky’s describes a “Zone of Proximal Development (ZPD)” [VYG-78] where students have partially formed learning of new concepts/skills, but are unable to successfully solve problems. He describes “cognitive scaffolding” as teaching and learning resources to support students solve problems while still in their ZPD – in this paper we argue that appropriately designed interactive computing learning tools can act as effective cognitive scaffolding

1.3 Related computer graphics education research

Existing work has been conducted in the form of computer-based learning environments and toolkits to support students of computer graphics. For example Grissom [GRI-96] describes a portable library of C++ routines to support 3D data structures and transformations. His approach was to initially use a third party tool to render the generated models, then to support students in writing their own rendering routines. However, such an approach “piggy-backs” the issue of students understanding of the mathematics of transformations, since being provided with a library of transformation functions, students will have little motivation (except final exam preparation) to learn the theory behind such library functions. Robbins [ROB-96] took an approach closer to the approach described in this paper – his “Tinker Toys Online” presented the user with several windows, allowing the user to view and transform 3D rendered models, view their transformation matrices, and view and edit 3D parameters such as clipping planes and projection reference point. However, Robbin’s system did not allow users to interactively edit transformation matrices, only to see how the matrices changed when mouse-driven transformations were requested by the user. Since the system described in our paper supports learning of three-by-three matrices for 2D transformations, it could well be argued that Robbin’s system would be a logical next step for students moving from 2D to 3D having used our LEG system.

2 Desired learning outcomes

An interactive learning environment has been developed to support students learning of the following:

- (i) To be able to describe the visual affect of basic geometric transformations (including translation, scaling, and rotation)
- (ii) To be able to predict the new position of each point of an object for a given transformation and given parameters (e.g. new coordinates after applying a translation of [-5, 10])
- (iii) To be able to describe and reproduce the three-by-three identify matrix
- (iv) To be able to describe the parts of the identify matrix that change when applying each basic transformation with given parameters
- (v) To be able to write the three-by-three matrix for a given transformation and given parameters
- (vi) To be able to perform with pen and paper the matrix multiplications appropriate for applying two-dimensional geometric transformations to a given point
- (vii) To be able to representing a sequence of transformations as a single matrix, having created and correct multiplied together matrices for each individual transformation in the sequence

3 The learning environment

The screen shots below illustrate the computer application that has been developed, called “LEG: Learning Environment for Graphics”. Each of its features, and how it integrates into a sequence of learning activities is presented in the next section.

As can be seen in Figure 1 below, the application has two main components, a graphics canvas on the left and a set of buttons and other interactive components on the right.

Figure 1: Screen shot of LEG (basic mode).

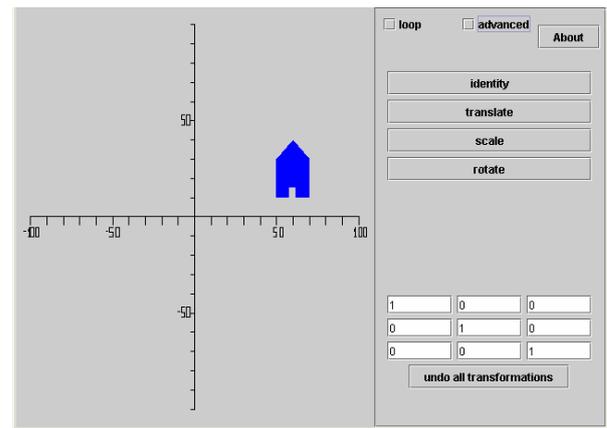


Figure 2 shows the set of simple transformation buttons (for translation, scaling and rotation), as well as other buttons and checkboxes:

- Button “identify” resets the displayed matrix to the identity matrix
- Button “About” displays some basic information about the application and author
- Checkbox “advanced” causes more advanced transformation buttons to be displayed (described in detail later)
- Checkbox “loop” repeatedly applies the current transformation matrix to the graphical object on the canvas (described in detail later)

Figure 2: The basic buttons.

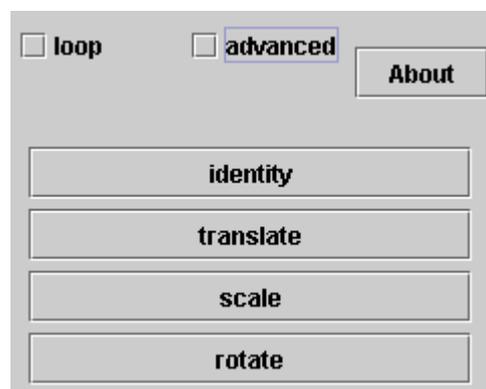


Figure 3 shows the graphics canvas comprising the left side of the application – it presents labelled X- and Y-axes

Figure 3: The graphics canvas.

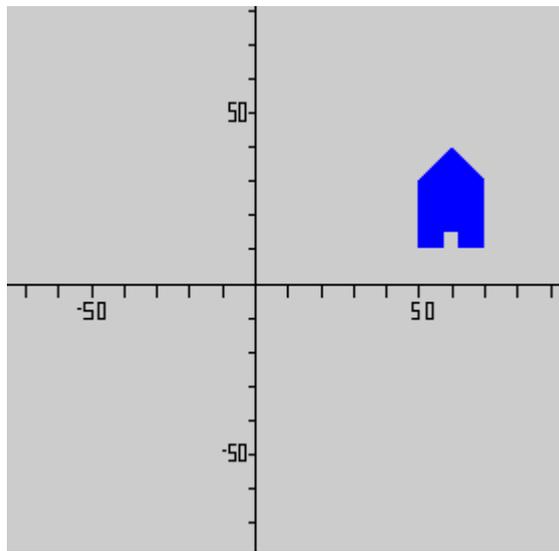


Figure 4 shows an example of the dialogs presented to the user requesting input of any parameters needed for the transformation button selected (in this example the user has clicked the “rotate” button so is being asked for the angle by which the graphic object should be rotated)

Figure 4: Dialog to input rotation parameter.

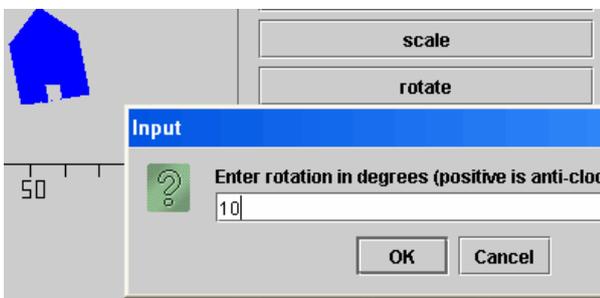
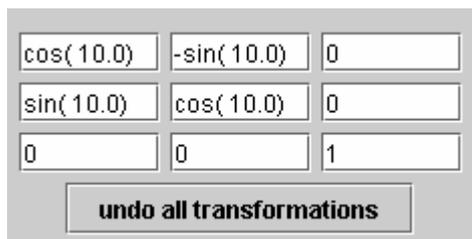


Figure 5 shows how the editable three-by-three matrix is updated after transformation parameters have been entered by the user. The values in this on-screen matrix are used to calculate the transformation of each point in the graphic object (house polygon) on the canvas.

Figure 5: Matrix created from rotation dialog input.



4 Progressively reducing student supports (scaffolding)

The strategy for using of the tool to support learning is straightforward. A consequence of a constructivist approach to computer-support learning is that students are provided with tools/computer environments to explore, some questions to answer and in the case of our LEG system, a recommended **sequence** of ways of using the tool.

The sequence students use the tool is designed so that students progress from **fully supported** by the system to **unaided** problem solving ability. By fully supported we mean that initially students having little or no understanding of the results or mechanics of transformation calculations will be passively viewing the transformed graphic object and learning the names of each transformation. They will completely rely on the “scaffolding” offered by the tool to support their understanding and ability to perform transformations on a graphic object. Having successfully used the tool students will (hopefully) be able to being able to describe and calculate graphical transformations with pen and paper – i.e. unaided by the LEG tool. Therefore they will have progressed to the point where they need no “scaffolding” to support their understanding and ability to perform transformations on a graphic object. This progressive reduction in the amount of support offered by LEG to students is illustrated in Table 1.

Table 1: Progressively reduced LEG scaffolding.

<i>Level of scaffolding</i>	<i>How LEG is used to transform 2D object</i>
MAXIMUM	Student passively views animated LEG transformations demonstrated by tutor
HIGH	Student clicks LEG buttons and entered parameters to apply transformations to object
MEDIUM	Student uses LEG button/parameter interface to investigate changes to LEG transformation matrices
LOW	Student directly edits LEG matrices to transform object
ZERO	Student works with pen-and-paper to define and apply transformation matrices

5 Use of the tool for learning

Small teams of students are given the list of desired learning outcomes, annotated with suggested ways of using the tool to achieve the outcome. It is then up to the students to work with the tool to discover and document their learning. After a suitable period the

class is stopped, and students asked to present their findings with explanations to justify what evidence they have to support their answers.

The annotated list of learning outcomes presented to students is as follows:

- (i) To be able to describe the visual affect of basic geometric transformations (including translation, scaling, and rotation)
 - Write down the name of each button on the tool, and explore the tool with different values to be able to write a description of what each button does
 - Identify which buttons affect the picture of the house on the left of the screen – for each write a definition of these transformations, e.g. “a scale transformation affects a graphical object by”
- (ii) To be able to predict the new position of each point of an object for a given transformation and given parameters (e.g. new coordinates after applying a translation of [-5, 10])
 - Note the coordinates of the bottom left corner of the house before and after applying each transformation
 - Person A should tell person B which button to press and any required parameter values – person B must first predict what the new coordinates of the bottom left corner of the house will be before performing the actions stated by person A
- (iii) To be able to describe and reproduce the three-by-three identify matrix
 - Every few minutes switch off the screen and write down the identify matrix – keep doing this until all team members can do this correctly every time
- (iv) To be able to describe the parts of the identify matrix that change when applying each basic transformation with given parameters
 - Person A should state which button to press and any required parameter values – person B must first predict which parts of the matrix will be changed – person C must predict what VALUES the changed parts of the matrix will change to
- (v) To be able to write the three-by-three matrix for a given transformation and given parameters
 - Person A should tell person B how they want the house to be transformed (try also stating exactly what the new coordinate of the bottom left of the house should be, and what direction the top of the house should point) – person B should edit the matrix to perform the stated transformation
- (vi) (not supported by the tool yet)
- (vii)(not supported by the tool yet)

6 Animating repeated transformations

A Java “Timer” is created when the “loop” checkbox is selected. This creates animated movement with a frame rate of 20 f.p.s. by repeatedly applying the current transformation matrix to all vertices of the canvas object (house). By choosing a **relatively** fast speed students are required to explore different parameter values. For example scaling with a factor of 2 means the graphic object moves beyond the canvas limits too quickly to see, therefore students need to try out different values such as 1.1 and 0.9 to transform the object by small amounts to see and confirm the results of repeatedly applying the same transformation.

7 Informal evaluation of the tool

Initial, informal experiments with computer graphics novices have been encouraging. Team sizes of two and three students were created. Little or no support required from the tutor (apart from the annotated learning outcome list presented in section 4 above). After using the tool students were able to answer questions such as:

- What is a translation/reflection etc. ?
- Rotate the house 20 degrees clockwise/anticlockwise etc.
- What parameters are needed for a translations/reflection etc.
- What part of the three-by-three matrix is changed for a translation/reflection etc. ?
- What three-by-three matrix would be needed to rotate the house by 5 degrees anticlockwise

8 Conclusions & Further work

An interactive computer learning tool has been developed, driven by a constructivist approach to learning. The software tool has been informally evaluated with small teams of novice computing students, and results of the evaluation were encouraging.

The tool illustrates a form of “cognitive scaffolding” – students can use the tool to perform graphical transformations with no knowledge of the calculations and matrices involved (through button and parameter interactions), and then progress to directly editing the transformation matrix and applying that matrix to the visible graphical object. Mistakes and unexpected results can be validated against matrices generated from the buttons and parameters. It is intended that after using the tool for some time students “wean” themselves off the need for the button/parameter interface and confidently edit the on-screen matrix alone. Having achieved the learning outcomes students then have no need for the computer tool at all, (hopefully) confidently and correctly creating and calculating geometric transformations with pen and paper.

While the tool appears to support learning of outcomes (i) .. (v), it does not support learning of outcomes (vi) and (vii). The learning of basic matrix multiplication (vi) will (hopefully) be improved after use of the LEG tool, since students will be able to relate the mechanics of such mathematics to its relevance for computer graphics – informal experience supports this assumption. The tool could be extended to support (vii) – combination of a sequence of transformations as a single three-by-three matrix, and such extensions are being explored. One metaphor being explored is an animation “timeline”, similar to those found in multimedia authoring tools such as Macromedia Flash, making clear the importance and difference in results of a multiple transformations performed in a particular order.

Acknowledgements

This work was part funded by a “learning innovation grant” from Middlesex University, London, UK.

References

- [GRI-96] GRISSOM S. "uisGL: A C++ Library to Support Graphics Education," *Computer Graphics*, 30 (3), August 1996.
- [HOL-89] HOLLAND S. 1989. *Artificial Intelligence, Education and Music*, Unpublished PhD thesis, IET, Open University, UK, 1989.

[LST-99] LAWS P., SOKOLOFF D. & THORNTON R. 1999. "Promoting Active Learning Using the Results of Physics Education Research". *UniServe Science News* Volume 13 July 1999, The University of Sydney, Australia.

[MAT-88] MATTHEWS M.R. (Ed.). 1998. *Constructivism and Science Education: A Philosophical Examination*, Kluwer Academic Publishers.

[ROB-96] ROBBINS A. “*Tinker Toys Online*”, unpublished Masters project, Computer Science Department, Rochester Institute of Technology, USA, 1996.
URL: <http://www.cs.rit.edu/~ncs/tinkerToys/>

[TIM-02] THE SUNDAY TIMES University Guide, September 15, 2002, "Ireland: Choose with care to stay the course"
URL: <http://www.timesonline.co.uk/article/0,,3501-412631,00.html>

[VYG-78] VYGOTSKY L. S. 1978. *Mind in Society: The Development of Higher Psychological Processes*, Harvard University Press, Cambridge, MA, USA.

Appendix – Oversize figures

Figure 6: Fifty degree anticlockwise rotation transformation – animated (“loop” ticked)

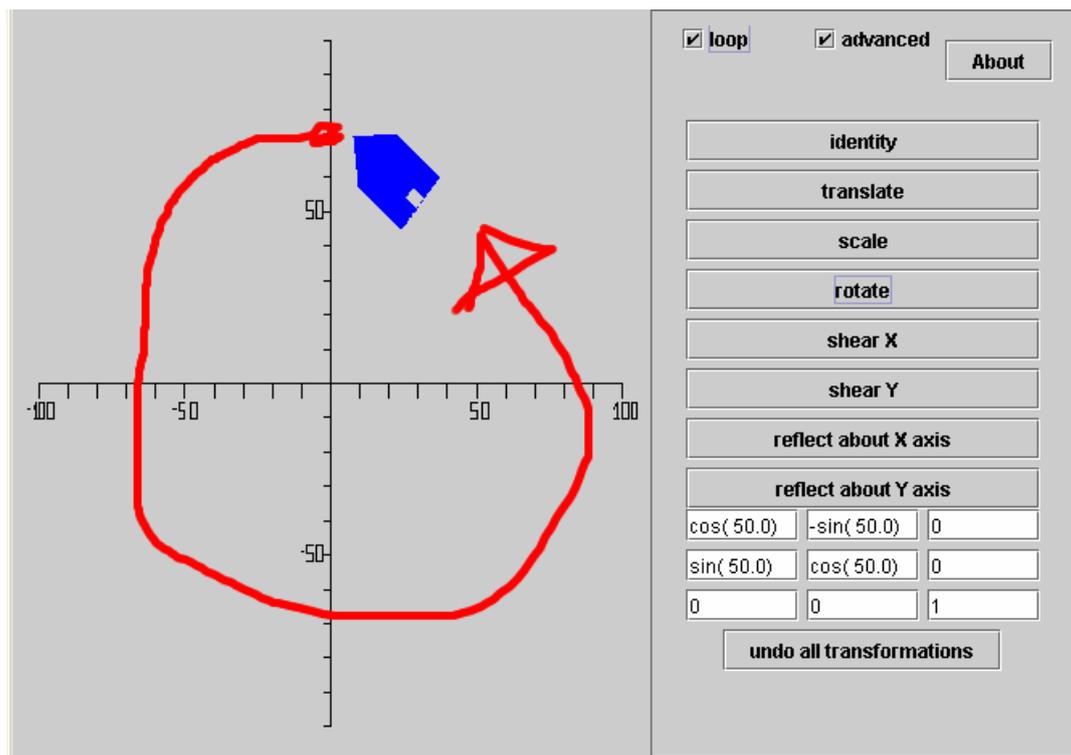


Figure 7: Application of a 0.5 Y-shear transformation to basic house object.

