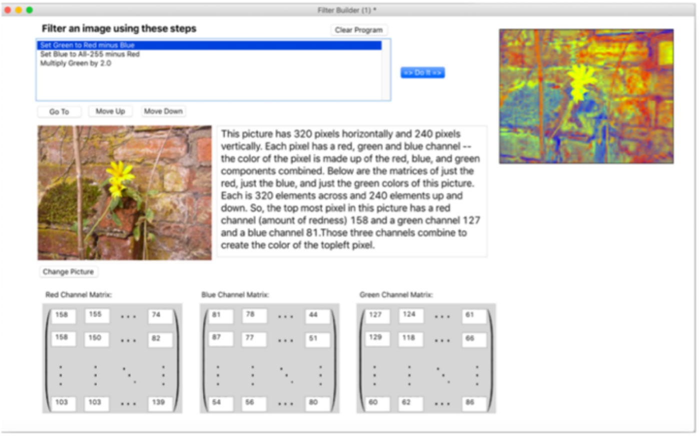
ARTS ENGINE

Creating Highly-Usable and Rapidly-Learnable Programming Language

Programming can provide motivating applications to engage students with learning in other subjects like pre-calculus and history, but existing programming languages are too hard. We work with teachers to design programming languages specific to tasks in secondary school classrooms, to make them adoptable, usable, and useful for students and teachers.



Filter an image using these steps

- Set Green to Red minus Blue
- Set Blue to All-255 minus Red
- Multiply Green by 2.0

This picture has 320 pixels horizontally and 240 pixels vertically. Each pixel has a red, green and blue channel -- the color of the pixel is made up of the red, blue, and green components combined. Below are the matrices of just the red, just the blue, and just the green colors of this picture. Each is 320 elements across and 240 elements up and down. So, the top most pixel in this picture has a red channel (amount of redness) 158 and a green channel 127 and a blue channel 81.Those three channels combine to create the color of the topleft pixel.

Red Channel Matrix:

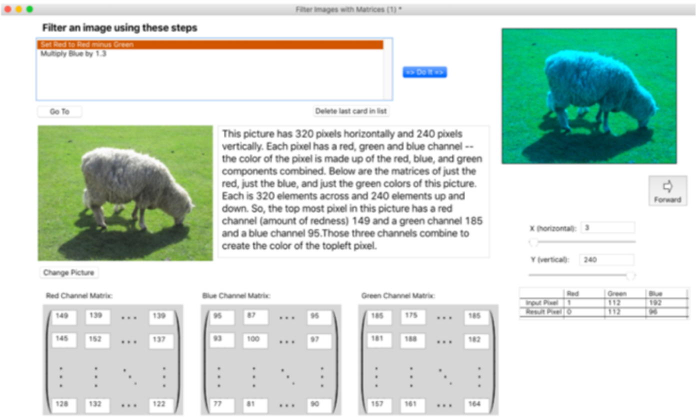
$$\begin{pmatrix} 158 & 155 & \cdots & 74 \\ 158 & 150 & \cdots & 82 \\ \vdots & \vdots & \ddots & \vdots \\ 103 & 103 & \cdots & 139 \end{pmatrix}$$

Blue Channel Matrix:

$$\begin{pmatrix} 81 & 78 & \cdots & 44 \\ 87 & 77 & \cdots & 51 \\ \vdots & \vdots & \ddots & \vdots \\ 54 & 56 & \cdots & 80 \end{pmatrix}$$

Green Channel Matrix:

$$\begin{pmatrix} 127 & 124 & \cdots & 61 \\ 129 & 118 & \cdots & 66 \\ \vdots & \vdots & \ddots & \vdots \\ 60 & 62 & \cdots & 86 \end{pmatrix}$$



Filter an image using these steps

- Set Red to Red minus Green
- Multiply Blue by 1.3

This picture has 320 pixels horizontally and 240 pixels vertically. Each pixel has a red, green and blue channel -- the color of the pixel is made up of the red, blue, and green components combined. Below are the matrices of just the red, just the blue, and just the green colors of this picture. Each is 320 elements across and 240 elements up and down. So, the top most pixel in this picture has a red channel (amount of redness) 149 and a green channel 185 and a blue channel 95.Those three channels combine to create the color of the topleft pixel.

Red Channel Matrix:

$$\begin{pmatrix} 149 & 139 & \cdots & 139 \\ 145 & 152 & \cdots & 137 \\ \vdots & \vdots & \ddots & \vdots \\ 128 & 132 & \cdots & 122 \end{pmatrix}$$

Blue Channel Matrix:

$$\begin{pmatrix} 95 & 87 & \cdots & 95 \\ 93 & 100 & \cdots & 97 \\ \vdots & \vdots & \ddots & \vdots \\ 77 & 81 & \cdots & 90 \end{pmatrix}$$

Green Channel Matrix:

$$\begin{pmatrix} 185 & 175 & \cdots & 185 \\ 181 & 188 & \cdots & 182 \\ \vdots & \vdots & \ddots & \vdots \\ 157 & 161 & \cdots & 164 \end{pmatrix}$$

X (horizontal): 3

Y (vertical): 240

	Red	Green	Blue
Input Pixel	1	112	192
Result Pixel	0	112	96

The long-term goal of this project is to provide access to computing education to all students (the "CS for All" movement, as coined by the Obama administration). The problem is that we have discovered that "access" isn't the problem. In many states, nearly half of all high schools offer CS classes, but there is no state where more than 5% of the students take any classes. If we want students to even see the possibilities of programming and computer science, we have to make it fit into other classes. An example: ANY history class in the United States is more diverse than the average CS class. The ratio of Black students taking AP CS Principles to those in AP US History is 1:5. For Hispanic students, it's 1:14. AP CS Principles is 24% female. AP US History is 56% female. If we can create programming tools that History teachers find useful and will adopt, we will dramatically increase the number and diversity of US high school students who get the experience of trying programming.

The UARTS Faculty Engineering/Arts Student Team (FEAST) will be tasked with developing and evaluating task-specific programming language prototypes for use in integrating computing into high school and undergraduate classes. We have experience in creating programming languages that are highly usable and learnable, e.g., users are typically working successfully in less than 10 minutes after starting. During the upcoming 18 months, the team will assess several already-developed prototypes. Team members will run participatory design sessions where mathematics and social studies teachers are asked to interact with several prototypes, so that they can give us concrete feedback on what works, what doesn't, and what they would really like. Based on this feedback, the team will refine the prototypes to improve usability and performance.

Meeting Time and Location

Mondays, 4-6pm

STUDENTS SOUGHT

USER INTERFACE DESIGN (2)

- Student skills: ability to sketch user interfaces or user scenarios, desire and interest in interviewing teachers and students (potential users), can help design and run participatory design sessions, can review notes from interview and focus group sessions to help identify needs and constraints on future designs, information visualization, typography, design drawing, sci-fi prototyping
- Likely majors: ARTDES, SI, ANTHRARC, COGSCI, COMM, EDUC, HISTORY, MATH, STATS, BCN, NEUROSCI, GS

USER INTERFACE PROGRAMMING (3)

- Student skills: front-end design and implementation interest, JavaScript and UI framework experience a plus but not a requirement, willing to program quick and dirty prototypes for use with teachers and students in participatory design sessions
- Likely majors: CS, SI, COMM

DIGITAL MEDIA (2)

- Student skills: familiarity with data in digital media (e.g., any of RGB/HSV color implementations, sound sampling, MIDI), interest in data sonification and visualization
- Likely majors: ARTDES, PAT, SI, COMM, PHYSICS, AMCULT

EDUCATION ASSESSMENT (1)

- Student skills: can help design and run participatory design sessions, can review notes from interview and focus group sessions to help identify needs and constraints on future design
- Likely majors: SI, EDUC, MATH

FACULTY PROJECT LEAD

Mark Guzdial is a Professor in Computer Science & Engineering and Engineering Education Research at the University of Michigan. He studies how people come to understand computing and how to make that more effective. He was one of the founders of the International Computing Education Research conference. He was one of the leads on the NSF alliance "*Expanding Computing Education Pathways*" which helped US states improve and broaden their computing education. He invented and has written several books on the "Media Computation" contextualized approach to computing education. With his wife and colleague, Barbara Ericson, he received the 2010 ACM Karl V. Karlstrom Outstanding Educator award. He is an ACM Distinguished Educator and a Fellow of the ACM. His most recent book is *Learner-Centered Design of Computing Education: Research on Computing for Everyone* (Morgan & Claypool, 2015). He received the 2019 ACM SIGCSE Outstanding Contributions to Education award.