

CHAPTER 4

Creating a Mosaic

Two-Dimensional Design

The first three chapters worked in continuous lines. A photograph became a silhouette, the silhouette was traced into a cut path, and the die cutter followed that path in a single unbroken stroke. Whether the result was a flat collage or a folding shadowbox, the language was the same: the smooth, mathematical curve of the vector outline.

This chapter introduces the language of individual picture elements (pixels). The previous three chapters uses vectors consisting of an origin, direction, and magnitude. Each method of creating computer graphics - vectors vs pixels - has its own advantages and disadvantages. A mosaic does not describe an image with a line at all. It builds the image from many small, separate units of color — tiles, beads, or bricks — arranged on a grid. Up close, only the separate squares are visible; from a few steps back, they combine into a crowned crane.

The concept of a matrix is central to digital design. It is how every photograph on a screen is stored, and it is the building-block thinking the next chapters extend into three dimensions. The mosaic is where it first takes physical form.

Bead mosaics are a craft with deep roots in many cultures. The activities in this chapter focus on designing a mosaic constructed from LEGO tiles.

The Logic of the Mosaic

Two ways that a computer can store an image - in vector or bitmapped formats - were introduced in Chapter 1. A mosaic stores each individual element separately rather than using a vector-based format. The Macintosh transformed design through introduction of a graphical user interface. The Macintosh included two graphical design programs: a paint program and a drawing program. The Macintosh paint program (MacPaint) was developed by Bill Atkinson and released with introduction of the Macintosh in 1984. The ability to use a graphical user interface was transformative in terms of art and graphical design.

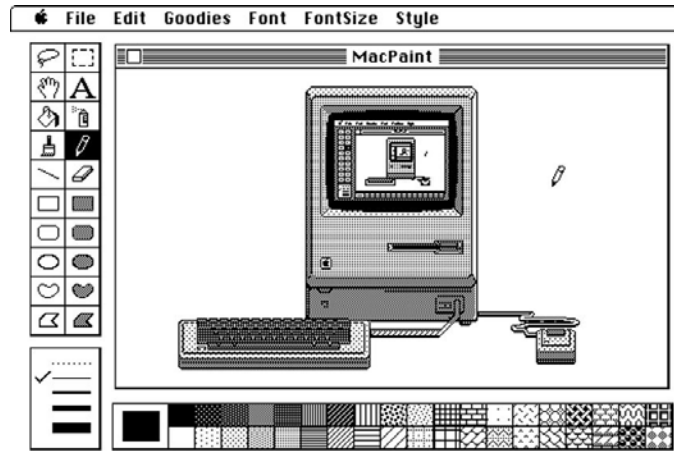


Figure 4.1. The Macintosh Paint Program (Bill Atkinson, 1984).

Although these programs appear to do similar things on the surface, dramatically different things occur inside the computer. Drawing programs use vector information. A line is described in terms of its length and direction. Drawing the same four lines requires two pieces of information for each line.

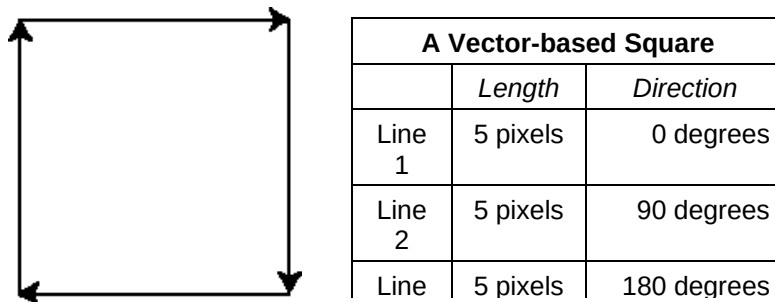


Figure 4.2. A Vector-based Square

In contrast, paint programs employ a bit-mapped format (Figure 4.3). The computer screen consists of a series of picture elements (*or pixels*). Paint programs record the status of each pixel in the memory of the computer. For example, a square drawn in a 10 x 10 block of pixels (left image below) is recorded as a “1” (black) or “0” (white) for each pixel. This drawing therefore consumes 100 bits of memory.

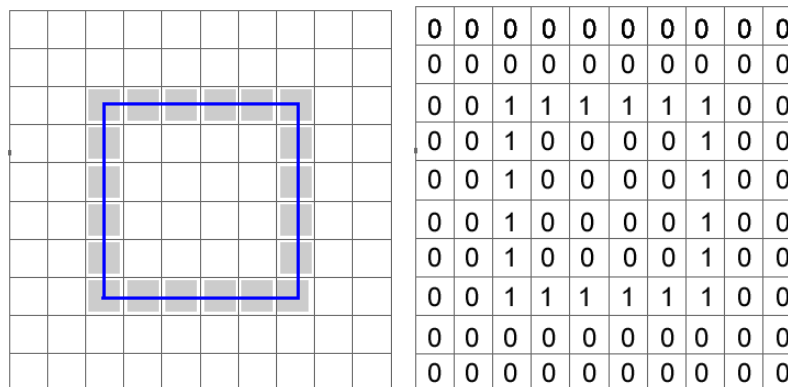


Figure 4.3. A Bit-mapped Representation of a Square

A mosaic is a bitmap built from physical pieces. Each tile, bead, or brick is one pixel — one cell of the grid holding one color — and the image is made by deciding, cell by cell, which color goes in each. Up close the separate squares are easy to see. From a few steps back, the pattern resolves as a recognizable shape. The image of the crowned crane on the left is composed of 278 x 413 pixels. If the image is reduced to ten percent of its resolution, the individual pixels can be seen, as shown on the image on the right.



Figure 4.4. A crowned crane (left) and low-resolution version with visible pixels (right)

The computer records each of the individual coordinates and colors for each of the individual picture elements. The lower-resolution image can be used as a guide to creation of a physical mosaic of the original crane.

Pixels and Vectors

Each way of storing a picture has a strength. A vector image is stored as instructions — the directions and lengths of its lines and curves — so a computer can redraw it at any size without losing sharpness. That is what a die cutter or printer needs: an exact path to follow.

A bitmap image is stored as a grid of colored cells, or pixels, with one color recorded for each cell. It is how photographs and screens work, and how every mosaic works. Its weakness is the reverse of the mosaic's strength: enlarge a bitmap and the cells themselves begin to show. Chapters 1 through 3 worked in vectors; this chapter works in pixels.

Bead Mosaics

The idea of building a picture from small colored units is far older than any computer. Bead mosaics are a familiar craft in East Africa.

A bead mosaic is a good place to begin, because it is a pixel image in the most literal way. Each bead sits in its own spot on a grid and adds one unit of one color, and the picture appears only when all the beads are taken in together. Beadworkers were thinking in pixels long before the word existed.

The bead mosaic is a demonstration rather than a project — it is here to introduce the idea. A simple bead tray, a card punched with a grid of small holes that each hold one bead, makes the principle easy to see: the picture goes down one unit at a time, and the grid can be filled, emptied, and filled again. Seeing a bead image come together gives a concrete feel for the way in which a mosaic is constructed, and it continues a living craft in the surrounding community.

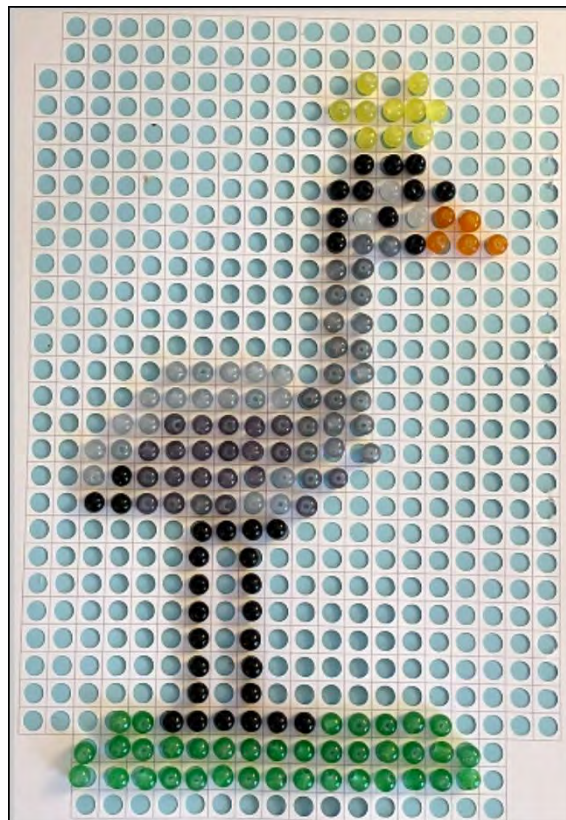


Figure 4.5. A bead mosaic of a crowned crane.

Ugandan Bead Art

Bead mosaics are made all over the world, and East and Central Africa have long traditions of the craft. In Uganda, much of this work is made by cooperatives of women. They arrange small glass or plastic beads — often rolled from recycled paper — into intricate patterns, in objects from jewelry to decorative panels. The craft is both artistic expression and livelihood.

Videos show this tradition directly: a demonstration of bead-mosaic technique and a visit with beadmakers at work. Watching them before the activity connects the grid on the table to the living craft it continues.

Beadworkers were thinking in pixels long before the word existed — one bead, one color, one cell at a time.

From Beads to Bricks

LEGO bricks make an excellent medium for prototyping. They can be combined with other components such as motors, actuators, and sensors to quickly develop iterations of a design — successive versions, each improving on the last. A LEGO baseplate is already a grid, and a 1 x 1 LEGO tile is already a pixel: one unit of one color that snaps into one cell.

The flat tile picture built here will become the starting point for a three-dimensional LEGO figure in the next chapter, and a 3D printed design after that. It is the first step of a path from two-dimensional to three-dimensional design.

The activity needs only a baseplate and a supply of LEGO tiles in a range of colors. A mosaic can consume many tiles of one color, so it is worth checking what is available before settling on a color scheme.

Sangala Mosaic: Using Design Tools

Creation of a low-resolution image with visible picture elements provides a useful starting point for design of a LEGO mosaic. Sangala Mosaic is a companion application to Sangala Studio. As described in the introduction above, the focus of Sangala Studio is vector-based graphics. In contrast, the focus of the companion application, Sangala Mosaic, is pixel-based (i.e., bit-mapped) graphics.

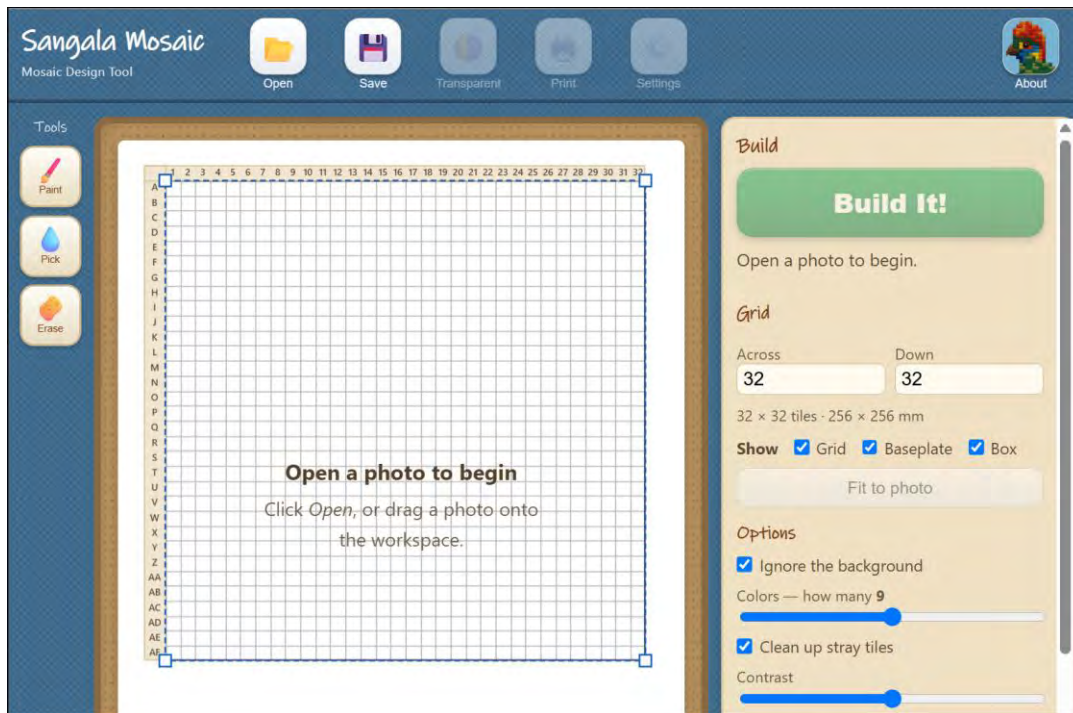


Figure 4.6. The Sangala Mosaic user interface.

The interface of Sangala Mosaic shares the same look-and-feel as Sangala Studio but focuses on editing the individual pixels of an image or photograph. At lower resolutions, these pixels can become tiles in a mosaic. Begin by dragging an image or photograph (a .jpg or .png file) into the workspace.

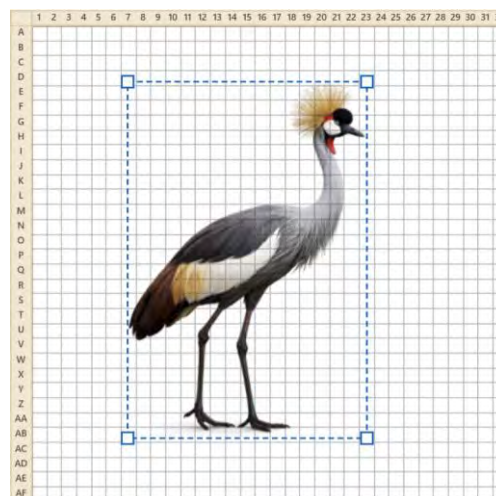


Figure 4.7. Placing the grid over an imported image.

A grid of cells provides an overlay that can be placed over the image. The default grid is 32 tiles wide and 32 tiles high because that is the size of the LEGO baseplate used to create LEGO mosaics in the examples below. However, you can change the number of

cells in the grid if you are working with a different number of tiles in the mosaic that you are building.

Working with as much resolution as possible makes the features of the object visible in the mosaic. This can be accomplished by re-sizing the image until it fills most of the frame. There is also a Remove Background button, as in Sangala Studio, that isolates a figure in a photograph.

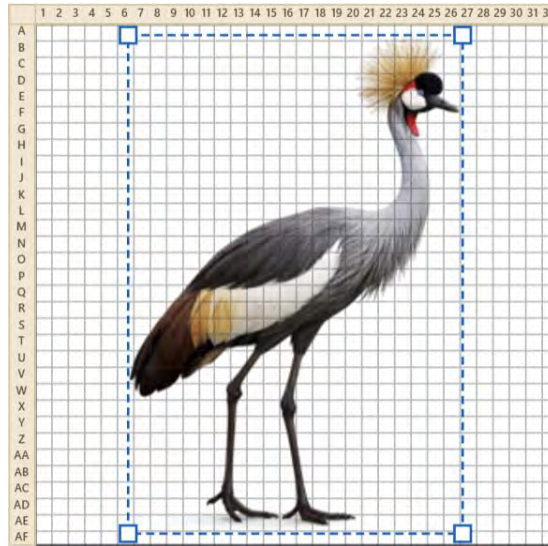


Figure 4.8. Resizing the image to fill the grid.

Sangala Mosaic provides a number of controls that adjust the tiles in the digital mosaic. Mosaic begins with a default set of tile colors but there are controls that specify the colors in Tile Colors — the set of tile colors that you want to use in building the mosaic. Once you have done that, use the Build It! button to translate the initial image into a mosaic.

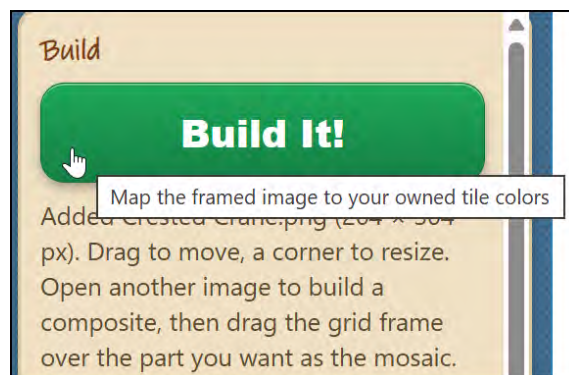


Figure 4.9. Building an initial mosaic.

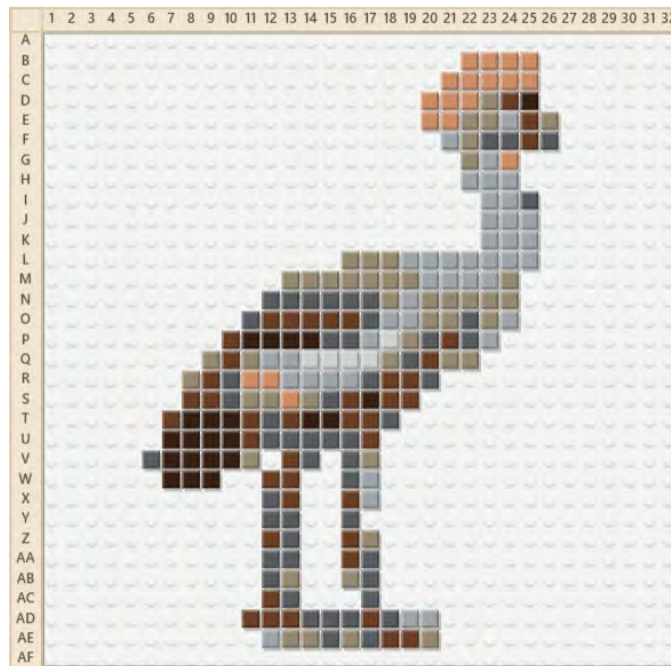


Figure 4.10. The initial design for a mosaic.

Sangala Mosaic translates the color of the image beneath the mosaic into a color for the tile above. The translation is constrained by the colors in the set of physical tiles available. The initial digital mosaic that results is a starting point for exploration. There are a number of controls that can adjust the tiles used in the *Adjust Mosaic* panel.

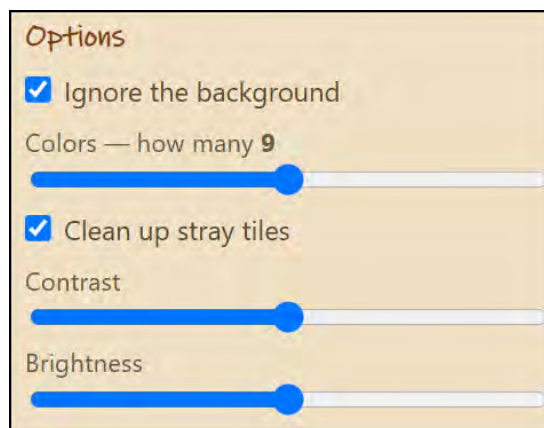


Figure 4.11. The initial design for a mosaic.

For example, reducing the tile colors can make features of the object stand out more clearly. The next example depicts the crane in two colors – gray and brown. Certain features stand out in this view: the body, the wings, the head, the bill, and the legs.



Figure 4.12. The initial design for a mosaic.

However, the designer also brings judgment to the task. The crest is a distinguishing feature of the crested crane – this feature is even included in its name – and that is not brought out by the draft mosaic design.

Contrast and brightness controls also adjust the view of the digital mosaic to gain insight into how it might be built. There is also a paint tool that assigns a color to a specific tile and an erase tool that removes stray tiles.

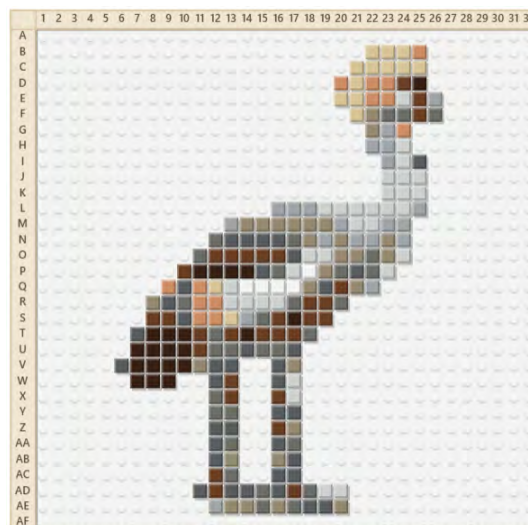


Figure 4.13. An adjusted view of the digital mosaic.

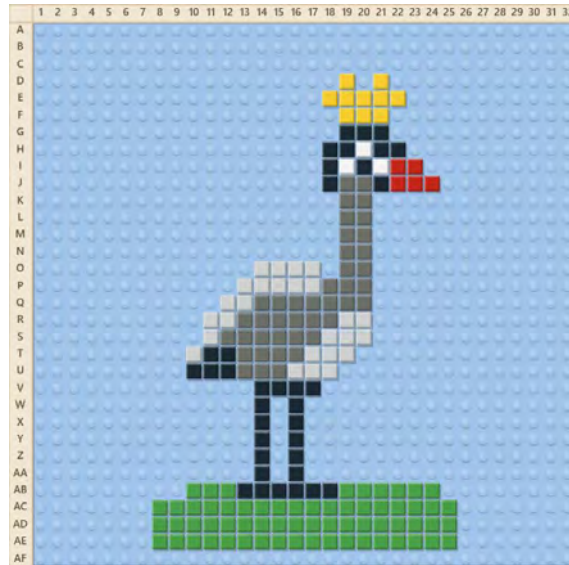


Figure 4.14. An adjusted view of the digital mosaic

Several tools can facilitate identification of masses — regions of tiles that belong to one part of the figure — that logically should be identified as a common element. The Select Area tool selects a group of tiles. In the case of the stylized example above, the crown, the bill, the legs, the wings, and the body could all be treated as elements. Once they are selected, the designer groups them and gives them a name.

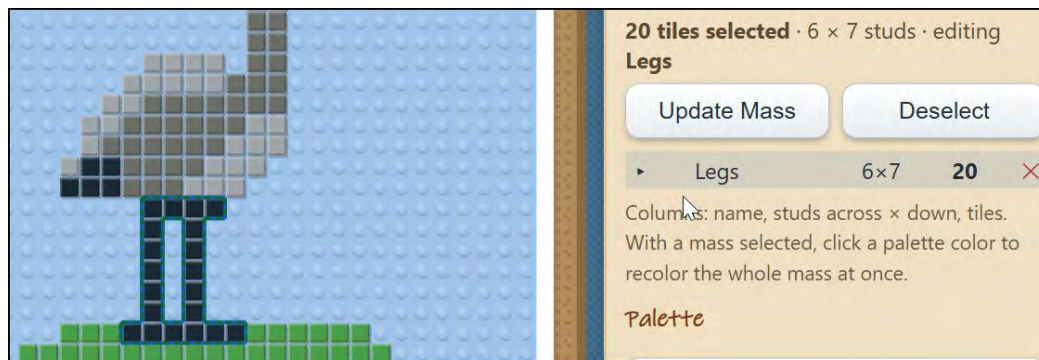


Figure 4.15. The legs have been selected and saved as a group named “Legs”.

A single click on the color palette gives all of the tiles in a group a common color. Grouping tiles in this way makes the transition from a mosaic that is a faithful translation of the colors of the underlying photograph to a more stylized version that captures the essential elements or underlying qualities of the object.

In engineering work, the term “iterations” is used to describe a series of successive designs that build upon and improve upon previous versions. This back-and-forth process — an interaction between designer and tools — is central to the method, and a small version of the collaboration that runs through the Sangala initiative.

Reading the Design on the Grid

Whether a design comes from a digital tool or a hand sketch, the same handful of rules determine whether it will read well as tiles — whether the subject is recognizable.

The first rule is boldness. At this size, fine detail disappears and strong, simple shapes remain recognizable. What to keep is the most recognizable silhouette — the fan of the crane’s crest, the long neck, the set of the legs. The second is contrast: neighboring areas must differ enough in color to be told apart, or a dark crane on a dark field simply vanishes. The third concerns curves: on a grid a diagonal becomes a staircase of single squares, and a smooth curve needs more cells to look smooth than a sharp angle does. A curve squeezed into too few cells just looks like steps.

If the crest is too faint, the next iteration calls for a bolder design; if two colors blur together, the designer swaps one out.

Designing on a Grid

Grid-based design has a long history in textile art, tile work, and needlecraft — any medium where the artist works unit by unit rather than with a continuous stroke. The same principles apply across all of them.

A small grid is forgiving precisely because every choice is visible. Working at this scale builds an intuition for composition in discrete units that applies directly to digital image editing, game-sprite design, and anything else built from pixels.

Translating Representational Images into Symbolic Art

Pixel art is image-making where the individual pixel is the unit of composition — every pixel placed deliberately, usually on a small canvas (16 x 16, 32 x 32, 64 x 64) with a tightly limited color palette. It grew out of hardware constraints in 1980s and early-90s computers and consoles, where memory limits capped resolution and colors. It survived those constraints and is now a chosen aesthetic. Minecraft is one of the best-known examples in which this is a deliberate choice but retro games such as Stardew Valley also embody pixel art.

At this scale detail cannot be rendered so it is suggested — a face might be five pixels, and an eye completes it. Every pixel is a decision with visible consequences, which makes it

unusually legible as a discipline. You can look at a good sprite — a small pixel-art figure — and see exactly what the artist chose.

Development of a LEGO mosaic made of flat LEGO tiles or a three-dimensional LEGO figure constructed from LEGO bricks is subject to similar constraints. LEGO only sells the flat LEGO tiles in 45 different colors, shown in the Sangala Mosaic palette below.

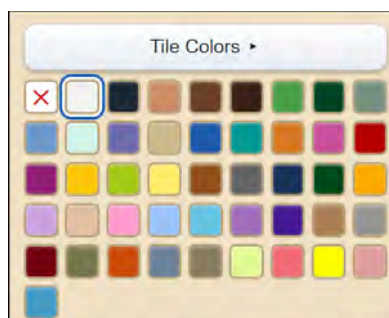


Figure 4.16. The Sangala Mosaic Palette (45 colors)

In addition, the most common LEGO baseplate consists of a 32 x 32 grid. Therefore, the process of translating an image such as a picture of a crane can benefit from the artistic insights gained through generations of exploration of pixel art.

The figure below shows the crowned crested crane at two levels of abstraction. The photorealistic (representational) version on the left depicts a literal translation of the pixels of the higher resolution photographic image into the closest colors available. The figure on the right is a symbolic abstraction of the representational version on the left. It translates the elements of the crane into distinctive features that will cause the viewer to recognize it as a crane.

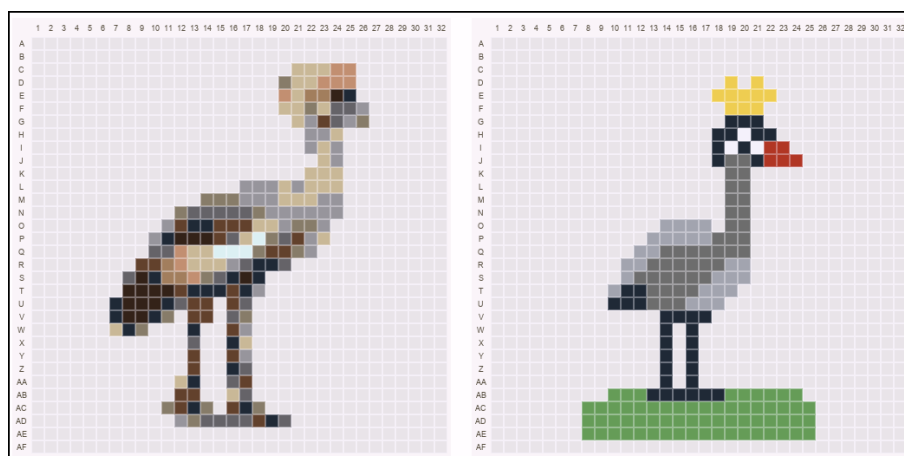


Figure 4.17. The Crowned Crane in Representational Form (left) and Symbolic Form (right)

The most distinctive feature of the crowned crane is the crown of feathers on top of its head. However, there is not sufficient resolution in a 32 x 32 grid to display the individual feathers of the crown so this feature is translated into a solid blob in the photorealistic version.



Figure 4.18. The Crown of Feathers in a Photograph (left) and as a Solid Blob in the Representational Version (right)

The most crucial stylistic decision required by pixel art is to treat this feature symbolically rather than literally. The indistinct blob – which contributes nothing to visual recognition of this feature – is replaced by a gold crown. The black of the head is replaced by black pixels. The eye is represented by three white pixels. The bill cannot be black – otherwise it would merge with the shape of the head. Treated symbolically, the traditional color for a bill is yellow, but that color is already taken by the yellow crown. Therefore, the designer uses red instead, to distinguish the bill from the head.



Figure 4.19. The Head and Crown Treated Symbolically

In the process of creating the symbolic version, the amorphous blob of the head has been sharpened and compacted to make the features more distinctive. The process of symbolic translation continues with the neck. The head is seven pixels wide in both the representational and the symbolic version. However, the red bill occupies three of these pixels on the lower row. The designer reduces the neck from three pixels to two, so that the head extends past the neck on each side.

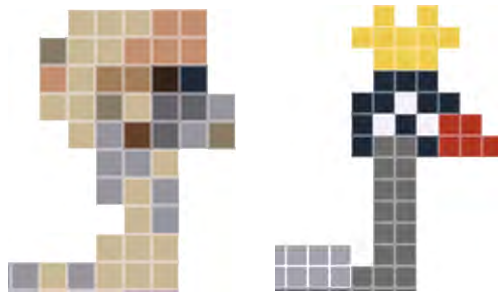


Figure 4.20. The Neck in the Representational Version (left) and the Symbolic Version (right)

The process continues with the body. Making the wing dark gray and the body light gray sharpens the distinction between them. Removing a pixel between the body and the neck sharpens the contrast between them.

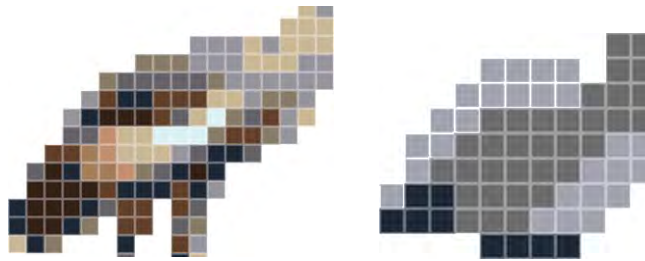


Figure 4.21. The Body and Wing in the Representational Version (left) and the Symbolic Version (right)

The legs are given a similar treatment. The designer collapses the legs into two columns of black tiles, which later will become two columns of LEGO bricks in the three-dimensional LEGO figure. A field of green below the legs represents grass.

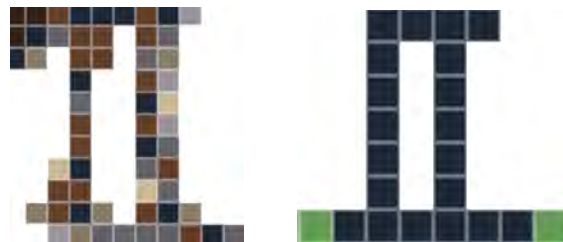


Figure 4.22. The Legs and Feet in the Representational Version (left) and the Symbolic Version (right)

In this way, the representational version of the crane is translated into a symbolic version that is suited to the reduced color palette and resolution of LEGO bricks.

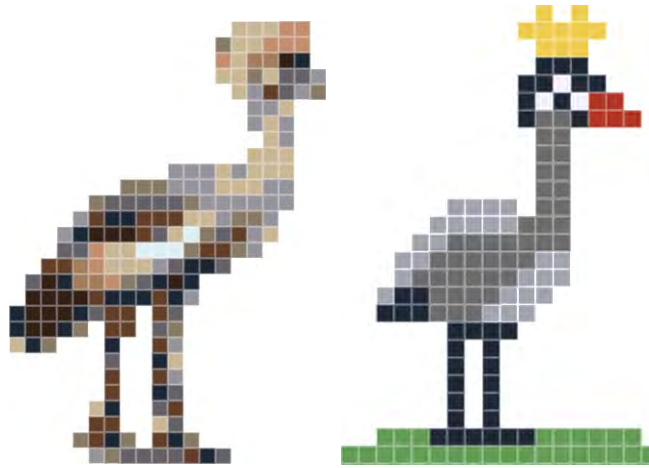


Figure 4.23. The Representational Crane (left) and the Symbolic Crane (right)

The final LEGO tile mosaic will then provide a framework in which the parts can be extruded to create a three-dimensional LEGO figure. *Note:* The process of translating the flat LEGO tile mosaic into a figure of three-dimensional LEGO bricks generally works best with a side or profile view, as opposed to a top or front view.

Building the Mosaic

With a satisfactory design in hand, building the mosaic is calm, methodical work.

Work from the design, filling the grid a region or a row at a time so that finished areas are not jostled as new tiles go in. Because tiles lift off cleanly, nothing here is permanent: a color that looked right in the design but wrong in the brick can be pulled and replaced.

Step back often. A mosaic is meant to be read from a distance, and a shape that looks clear from a few inches away can be unreadable across the room. Stand the baseplate upright, step back, and check whether the crane reads; then adjust contrast, proportion, and silhouette until it does.

Another refinement could involve use of LEGO tiles that do not have the studs (i.e., round pegs on top) that are found on most LEGO bricks. The studs are used to attach an additional layer of LEGO bricks to the layer below. In this case, the mosaic is complete in itself and does not require additional bricks. Since that is the case, it is possible to use flat LEGO tiles without the studs on top.



Figure 4.24. Flat LEGO tiles on a green background.

When the mosaic is complete, photograph it straight on. That photograph is not just a keepsake. In the next chapter it becomes the reference for building the three-dimensional crane, so the clearer the flat design, the easier the three-dimensional one will be.

Designing Bead Mosaics

The LEGO tile mosaics developed with Sangala Mosaic will be used as the starting point for three-dimensional LEGO figures in the next chapter. However, there are many other mediums in which mosaics can be constructed.

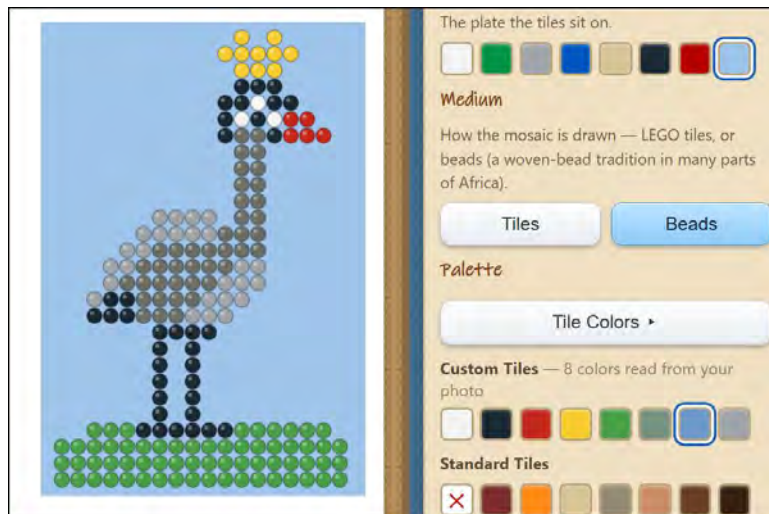


Figure 4.25. Representing the mosaic in the medium of beads.

The Medium control in Settings displays the design as tiles, dots, or beads.

Example of a LEGO Mosaic

The following example of a tiled LEGO mosaic was initially developed using Sangala Mosaic as a design tool.



Figure 4.26. The initial design created in Sangala Mosaic.

Moses Kumenya, the Sangala Innovation Teacher, then used this digital design as the inspiration for a physical implementation using LEGO tiles.



Figure 4.27. The physical mosaic constructed with LEGO bricks.

Note that the physical version is not an exact copy of the digital design. Moses Kumenya added other details, such as the clouds and grass.

Julia Skinner and Claire Mills chose a side view of a hippopotamus, judging that the profile carried more of the animal's character than a front view would. The round, heavy body proved the hardest thing to state in square tiles — a direct lesson in the tension between a curved subject and a rectangular grid.

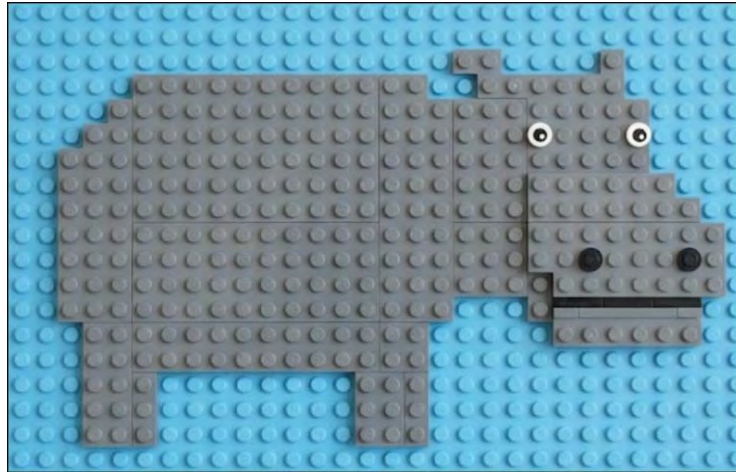


Figure 4.28. LEGO mosaic of a hippopotamus (Julia Skinner & Claire Mills)

Their own accounts, and further examples, can be found in the Educational CAD Model Library forum.

Looking Ahead

In the next chapter the mosaic becomes a plan for a three-dimensional LEGO crane, turning the logic of the pixel into the logic of the building block. That sculpture in turn becomes the model for a digital design and a 3D print in the chapters that follow.

The crane began as a photograph, became a silhouette, and grew into a collage and a shadowbox. Now it returns as a grid of colored tiles. Each material presents the same subject with a different problem.