Into the Shadows:
Approximating Images by Orthogonal Projection

Kelly Delp
Mathematics Department
Ithaca College
kelly.delp@gmail.com

Sam Lloyd
Mathematics Department
Ithaca College
slloyd1@ithaca.edu

Abstract

We experimented with approximating images by projecting them onto the span of a relatively small set of matrices that correspond to images. By decomposing our images into 3 or 4 sub-images, we were able to obtain recognizable approximations in a subspace with dimension less than 10% of the original matrix size.

Introduction

Our explorations in this paper were inspired by the short film by Hector Rodriguez, *Theorem 8*, which was shown in the 2014 Bridges film festival [3]. In this film, Rodriguez approximated a segment of Godard's 1964 film "Alphaville" using 100 stills from another (never completed) film. The central mathematical idea explored in the film was this: $n \times m$ black and white images are vectors in $\mathbb{R}^{mn}$. Take the span, that is, the set of all linear combinations, of the 100 stills to form a 100-dimensional subspace of $\mathbb{R}^{mn}$. Then, take the orthogonal projection (appropriately rescaled) for each frame of the movie "Alphaville" to create a new movie whose frames are approximations of those in Alphaville.

We found the result of Rodriguez's project to be intriguing and somewhat surprising. Even with a relatively poor resolution of images, say 100×100, the images would lie in a vector space of dimension 10,000. A mere 100 stills would intuitively seem to be far too few to get any sort of recognizable approximation. Indeed, the individual stills in the approximation are very abstract and ghostly. However, the fact that these ghostly images were shown in tandem with the original film allows the viewer to see the relationship to "Alphaville". One advantage of the medium of film in this sort of analysis is in the motion; large-scale features of the approximations move in sync with the original.

There are many questions to be explored. Could one achieve any sort of interesting results with this process without motion? How does changing the collection of basis images, both in type and number, influence the approximations? If we use a basis of images that are similar in content to the target image do we get better results? Furthermore, how does using a basis of photographs compare to using a set of "random" images? In this paper we discuss some of these questions, as well as give a bit more detail about the mathematics involved. All of our computations were done in Mathematica, and the mathematics used is typically found in a first course in Linear Algebra. We think there is potential for sharing these ideas with students in such a course to illustrate the notions of projections, span, basis, and subspace.

The idea of combining hundreds of photos to approximate other images is not new. We'd like to thank Craig Kaplan for pointing out one example of such work by Jim Bumgardner, who took an unweighted average of 300 images of graffiti, chosen optimally from a store of 20,000, to approximate an image of the shroud of Turin [1]. While this paper and Bumgardner's work share the theme of approximating an image with a linear combination of other images, the method Bumgardner uses for creating this approximation is quite different from the orthogonal projection method used in *Theorem 8* and this paper.

Projections Onto Sets of Random Images

A black and white image can be represented by a matrix with entries lying in the interval $[0, 1]$, where the (i, j) matrix entry corresponds to the grayscale value of the (i, j) pixel. The value 0 represents black, and 1 white. Throughout this paper we will be working with matrices of dimension 100×100. The set M of all real-valued matrices of these dimensions forms a 10,000 dimensional vector space over $\mathbb{R}$, which we equip with the Frobenius inner product, $\langle A, B \rangle =$Trace$(A^T B)$. If we think of our matrices as a single list of numbers, this inner product is equivalent to the usual inner product on $\mathbb{R}^{10,000}$.
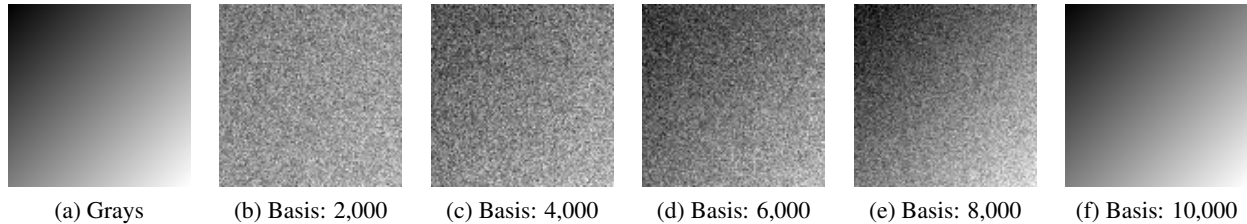

| (a) Grays | (b) Basis: 2,000 | (c) Basis: 4,000 | (d) Basis: 6,000 | (e) Basis: 8,000 | (f) Basis: 10,000 |

Figure 1: *Varying the dimension of the subspace*

Consider the image, Grays, in Figure 1a. Let $\{M_n\}$ be a set of 2,000 matrices with random real valued entries chosen uniformly from the interval $[0, 1]$. These matrices are almost certainly (with probability 1) linearly independent, and therefore the span forms a 2,000 dimensional subspace of $\mathbb{R}^{10,000}$, which we call W. Our goal is to find a matrix in W that best approximates the image Grays. The tool we use to do this is the following theorem, which is slightly paraphrased from Theorem 8, found on page 348 of [2]. (We suspect that Rodriguez may also use [2] as a reference.)

Theorem (The Orthogonal Decomposition Theorem). *Let W be a k-dimensional subspace of $\mathbb{R}^n$. Then each $\mathbf{y}$ in $\mathbb{R}^n$ can be written uniquely in the form $\mathbf{y} = \hat{\mathbf{y}} + \mathbf{z}$, where $\hat{\mathbf{y}}$ is in W and $\mathbf{z}$ is in $W^{\perp}$. In fact, if $\{\mathbf{v_1}, \ldots, \mathbf{v_k}\}$ is any orthonormal basis of W, then*

$$\hat{\mathbf{y}} = \langle \mathbf{y}, \mathbf{v_1} \rangle \mathbf{v_1} + \cdots + \langle \mathbf{y}, \mathbf{v_k} \rangle \mathbf{v_k}$$

and $\mathbf{z} = \mathbf{y} - \hat{\mathbf{y}}$.

For us, the subspace W is the span of the set of 2,000 random matrices and $\mathbf{y}$ the original matrix G corresponding to Grays . By using the Gram-Schmidt orthogonalization algorithm, we can transform the basis $\{M_n\}$ into an orthonormal basis, $\{O_n\}$, for W. We can then project the the vector G onto W by taking the inner product of G with each of the vectors O_n. These become the coefficients in our linear combination of the O_n. The projection, $\hat{G}$, is the best approximation (in terms of minimizing Euclidean distance) of G lying in the span of the original random image matrices. We then rescale $\hat{G}$ so that it's maximum entry is 1; the image corresponding to this rescaled $\hat{G}$ can be found in Figure 1b.

Figure 1 shows the results when we repeated this process with bases of varying sizes, including a full basis. Due to the size of these matrices and the basis, we were concerned about the stability of Gram-Schmidt. Mathematica has built in algorithms for Modified Gram-Schmidt as well as Gram-Schmidt, and allows us to choose our tolerance. The vectors we get out of the orthogonalization process are not quite orthogonal; however, we can set the tolerances so that the dot products of distinct vectors in the transformed basis are on the order of, say, 10^{-17}. The image in Figure 1f reassured us that this amount of error was acceptable.

Using a Basis of Photographs

The trials we ran with random images did not make us optimistic about getting good approximations of still images from relatively small sets of photographs. In our first trial, we tried approximating an image using a basis consisting of 200 images, appropriately cropped and resized to 100×100, from our personal photo libraries. Following the procedure outlined in the previous section, the results were unsurprisingly poor. The approximation did not yield a recognizable image. Examining the images from the previous section suggests a reason. Given a large enough $k \times k$ subgrid, one would expect a random matrix to have an entry close to both 1 and 0, making it difficult to get a large concentrations of black or white in the image. With photographs, we won't have this exact same problem, but it seemed that working with a basis in which the matrices had many zero entries could produce better results.
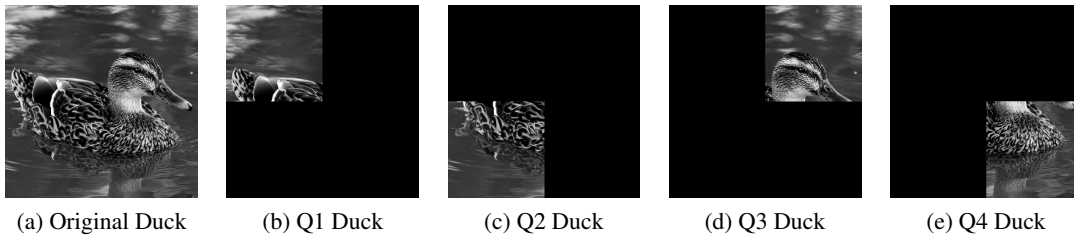

| (a) Original Duck | (b) Q1 Duck | (c) Q2 Duck | (d) Q3 Duck | (e) Q4 Duck |

Figure 2: *Quartered basis (Duck image cropped from a photo by Camilo Morales.)*

This was the motivation for an alternate method of basis construction. We started with our original 200 matrices, and chopped each of these into quarters as seen in Figure 2, thus creating four separate matrices for our basis. This new basis should still be linearly independent and also has the original images in the span, since Q1+Q2+Q3+Q4=image. Using these 800 matrices as a basis for the subspace yielded fairly good approximations, which can be seen in Figure 4 (a) and (d). However, looking closely at the image in Figure 4 (a), one can see a slightly quartered effect in the projection. We decided to try a different way of decomposing the images in our basis to avoid having visible horizontal and vertical mid-lines.
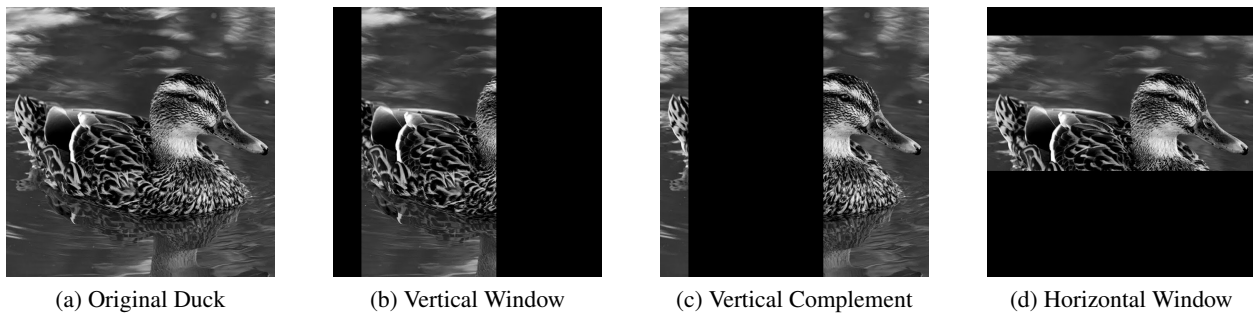

| (a) Original Duck | (b) Vertical Window | (c) Vertical Complement | (d) Horizontal Window |

Figure 3: *Window basis*

Our second method of basis construction involved decomposing our matrices into three partial images created by removing horizontal and vertical slices, as seen in Figure 3. Again, the original photos are in the span of these images. Including the complement of both the vertical and horizontal windows would lead to a linearly dependent set of matrices, as the sum of the vertical window matrix and its complement would be equal to the sum of the horizontal window and its complement, so we did not need to include both sets of complements. All windows have a width of 50 pixels. To avoid having very distinct vertical and horizontal line features in the projection image, we varied the location of the windows. In Figure 4 we show the results

of the projections using this basis. We were pleased with the quality of the projection given the fact that our basis was of size 600, generating a subspace that is only 6% of the dimension of the larger vector space.
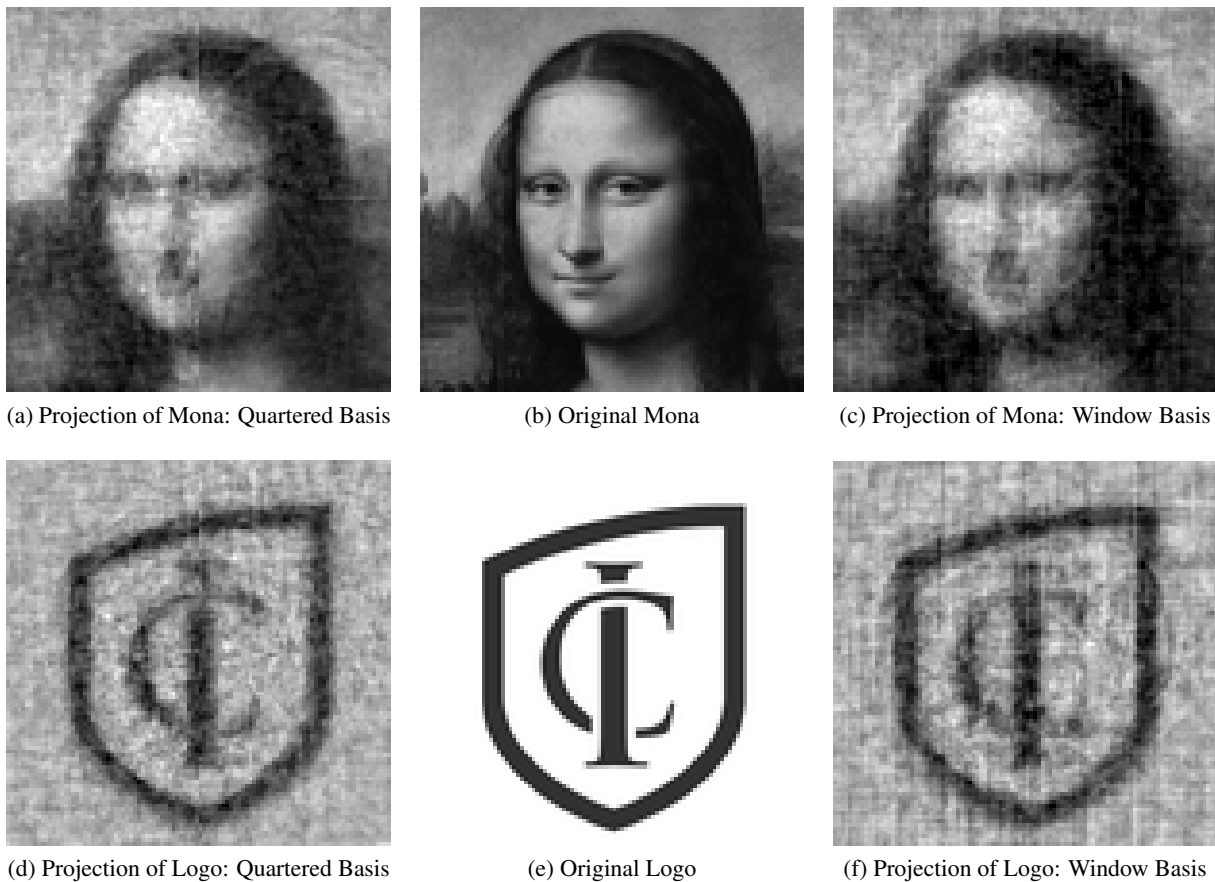

(a) Projection of Mona: Quartered Basis   (b) Original Mona   (c) Projection of Mona: Window Basis

(d) Projection of Logo: Quartered Basis   (e) Original Logo   (f) Projection of Logo: Window Basis

Figure 4: *Results of projecting onto the Window and Quartered Bases*

Decomposing the images into just a few sub-images provided a basis which yielded recognizable approximations. There are still several questions left to explore. For example, by restricting to images that are superficially similar to each other (eg., portrait photos) could we get good results without decomposing our basis? How many images would we need? Quantifying the error between the original and approximated image allows us to compare different methods. We have done a few preliminary computations of this nature which showed that the window basis provided better approximations than the quartered basis. We'd also like to experiment and compare results with other, carefully chosen bases; perhaps one of just stripes, or another containing matrices with blocks of white. For us, one of the most attractive things about this project is that it provides a visualization of ideas from linear algebra which we can then share with our students, family and friends.

References

[1] Jim Bumgardner. *Shroud of Turin.* Retrieved May 3, 2015, from
    http://krazydad.com/blog/2012/08/11/graffiti-shroud/

[2] David C Lay. *Linear Algebra and Its Applications*, 4th edition. Pearson, (2011).

[3] Hector Rodriguez. *Theorem 8.* Retrieved May 3, 2015, http://theorem8.concept-script.com/