AESvisual: A Visualization Tool for the AES Cipher

Jun Ma, Jun Tao
Department of Computer Science
Michigan Technological University
Houghton, MI
{junm,junt}@mtu.edu

Melissa Keranen
Department of Mathematical Sciences
Michigan Technological University
Houghton, MI
msjukuri@mtu.edu

Jean Mayo, Ching-Kuang Shene, Chaoli Wang
Department of Computer Science
Michigan Technological University
Houghton, MI
{jmayo,shene,chaoliw}@mtu.edu

ABSTRACT

This paper describes a visualization tool AESvisual that helps students learn and instructors teach the AES cipher. The software allows the user to visualize all the major steps of AES encryption and decryption. The demo mode is useful and efficient for classroom presentation and the practice mode provides the user with an environment to practice AES encryption with error checking. AESvisual is quite versatile, providing support for both beginners learning how to encrypt and decrypt, and also for the more advanced users wishing to see all the details, including the $GF(2^8)$ addition and multiplication operations. Classroom evaluation of the tool was positive.

Categories and Subject Descriptors

K.3.2 [**Computers and Education**]: Computer and Information Science Education—*Computer science education, information systems education*

General Terms

Algorithms, Security

Keywords

Cryptography, visualization

1. INTRODUCTION

In 1997, the National Institute of Standards and Technology asked for potential candidates to replace the Data Encryption Standard (DES) as the official data encryption standard. In 1998, there were five finalists, and eventually from this list Rijndael was chosen to be the winner. It was developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen [3]. The Advanced Encryption Standard (AES) is based upon Rijndael. It has been a federal government standard since 2002 and is now used worldwide.

AES is a type of block cipher. It consists of 10 rounds; each round has an input of 128 bits and produces an output of 128 bits. The algorithm has four basic steps, or layers, that when put together form the rounds. When studying this cipher, it is sometimes easier to focus on each step of the algorithm separately. Although each step is straightforward, students often have difficulties putting all of the pieces together. Therefore, when asked to complete one round of the algorithm, they may find they do not understand it in its entirety. We have created a visualization tool, AESvisual, to aid in the process of learning the cipher.

AES appears in nearly every cryptography and computer security textbook [6, 7]. Many tools are available ranging from some simple ones [2, 4] to publicly available and more sophisticated systems [1] such as applets directly accessible on the web. However, many pedagogical tools available now only provide an animation of the algorithm. The one that is closely related to our goal [5] uses hardware visualization. AESvisual is different in that it allows for the user to both view the process and practice using the cipher. Users can work through, in detail, each of the four layers: Substitute Bytes Transformation, Shift Rows Transformation, Mix Columns Transformation, and Add Round Key. The software also leads the user through the Key Expansion process, which is used to generate the key for the cryptosystem.

In the following, Section 2 provides the background of our cryptography course, Section 3 presents our visualization tool, Section 4 provides a detailed study of our findings from a survey, and Section 5 is our conclusion.

2. COURSE INFORMATION

AESvisual was used in a cryptography course, MA3203 Introduction to Cryptography, that is offered out of the Department of Mathematical Sciences at Michigan Technological University. It is a junior level course that gives a basic introduction to the field of cryptography. This course covers classical cryptography, the Data Encryption Standard (DES), the Advanced Encryption Standard (AES), the RSA algorithm, discrete logarithms, hash functions, and elliptic curve cryptography. For each cryptosystem, we study how it was designed, why it works, how one may attack the system, and how it has been used in practice.

The widespread use of AES makes it an essential algorithm for any introductory cryptography student to under-

Draft Department of Computer Science, Michigan Technological University, Houghton, Michigan USA, 2014

stand. Therefore, it is an important piece of our cryptography course, and we give a great deal of attention to it. AESvisual was used in the classroom to demonstrate an entire round of the algorithm throughly and efficiently. It was also used by students for self-study to learn and practice both the encryption and decryption processes.

3. SOFTWARE DESCRIPTION

AESvisual supports Windows, MacOS and Linux. It consists of two major components: the `Demo` mode and the `Practice` mode. The `Demo` mode displays both the encryption and decryption operations of the AES algorithm, and each operation has multiple pages to demonstrate the major steps. The `Practice` mode helps the user learn the detailed computations step-by-step and perform self-study. Only encryption is available in this mode since decryption follows the same workflow in a reversed order. A test report system helps the instructor verify student learning effectiveness.

3.1 The Demo Mode

AESvisual always starts from the `Demo` mode. It has four subpages: `Overview`, `Encryption`, `Decryption` and `Key Expansion`. The `Overview` subpage is used to demonstrate the workflow of the encryption and decryption operations and their relationship (Figure 1). Encryption and decryption involve ten rounds, but only the first round (highlighted in the red) is shown. Clicking the `Go` button below the `Round 1` marking brings the user to the `Encryption` subpage or `Decryption` subpage. The user may also click the `Expand Key` button to advance to the `Key Expansion` subpage.

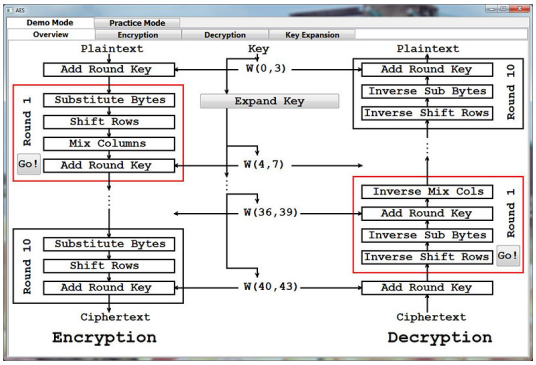

Figure 1: `Overview` of the AES Algorithm

3.1.1 *Encryption*

This subpage demonstrates the four major steps of the first round (*i.e.*, `Round 1`) for encryption: `Substitute Bytes`, `Shift Rows`, `Mix Columns` and `Add Round Key`. Each of these four steps has its own subpage.

Substitute Bytes. This subpage shows how the 128-bit original plaintext is processed using `Add Round Key` and the S-box transformation (Figure 2). The user may generate a new random plaintext-key pair with the `Random` button. The generated key is then expanded in the `Key Expansion` subpage to create 44 32-bit words. Clicking the `Expand Key` button brings the user to the corresponding subpage. The user may click the `Add Round Key` button to see how the plaintext is added with the first four words `W(0, 3)`. The output is then transformed with the S-box Transformation. The user may select an element (in red) in the output matrix of the `Add Round Key` subpage and then click the `Check S-box` button to see the details of the transformation (Figure 3). The corresponding element in the result is highlighted and the selected row and column are also shown above the `Check S-box` button. The result from this transformation is then used as the input matrix to the `Shift Rows` subpage.

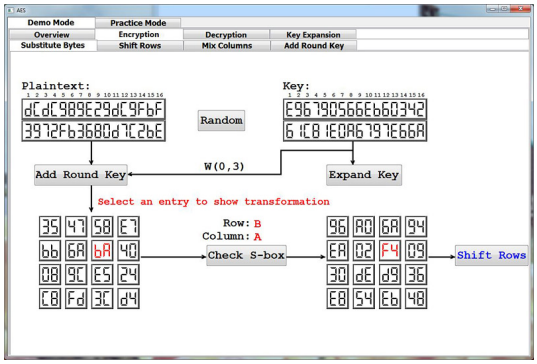

Figure 2: `Substitute Bytes` of `Encryption`

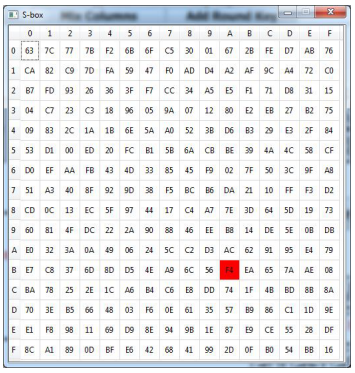

Figure 3: `SBox` for the S-Box transformation

Shift Rows. This subpage demonstrates how the input matrix is transformed by performing row-based byte rotation (Figure 4). The result goes to the `Mix Columns` subpage.

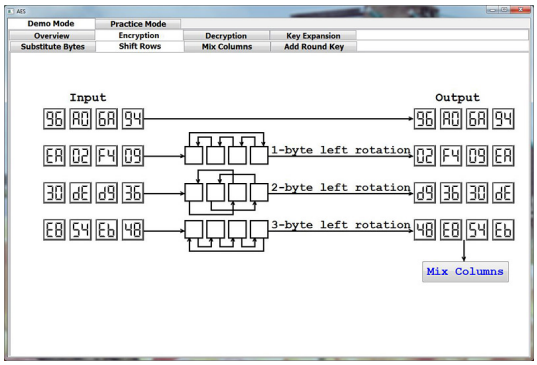

Figure 4: Shift Rows of `Encryption`

Mix Columns. This subpage shows how the output matrix is obtained by multiplying the input matrix with a given matrix in GF(2^8) (Figure 5). If the user selects a column (in red) of the input matrix, the corresponding column of the output matrix will be highlighted (in green). The lower half of this subpage has the details of the matrix multiplication for the selected column. The user may click $\times$ and $+$ to

explore the corresponding GF(2^8) multiplication (Figure 6) and addition (Figure 7) operations in detail. The numbers in these windows are in binary format. The output matrix is then used as the input for the `Add Round Key` subpage.

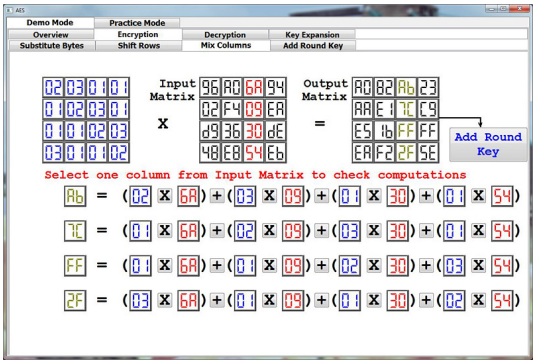

Figure 5: Mix Columns of `Encryption`

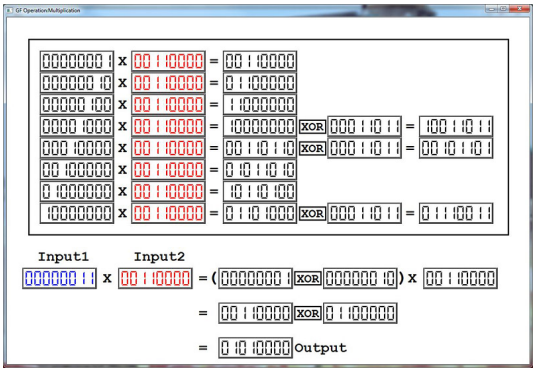

Figure 6: G(2^8) Multiplication

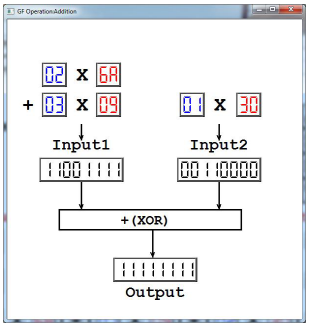

Figure 7: G(2^8) Add

`Add Round Key.` This subpage shows how the input matrix is XORed ($\oplus$) with the word matrix element-by-element (Figure 8). The user may select an element of the input matrix (in red) or the word matrix (in blue) and the corresponding element in the output matrix will be highlighted (in green). The lower half of this subpage shows the corresponding exclusive disjunction operation in binary format. The final ciphertext after the ten rounds of the encryption process is shown in the lower right corner of this page.

3.1.2 *Decryption*

The `Decryption` subpage also has four subpages showing the four major steps of the first round of decryption. It starts with the `Shift Rows` subpage (Figure 9), followed by `Substitute Bytes`, `Add Round Key`, and `Mix Columns`. The

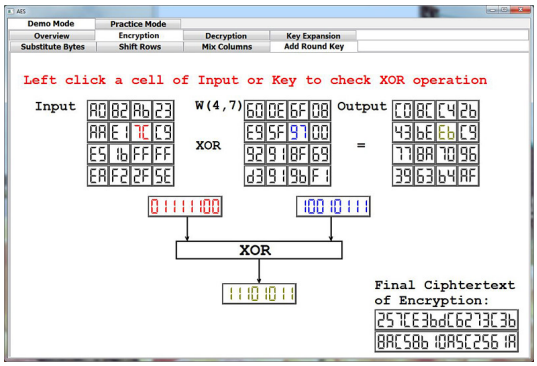

Figure 8: Add Round Key of `Encryption`

ciphertext in `Shift Rows` is taken from encryption and the user may click the `Add Round Key` and `Substitute Bytes` buttons to advance to the corresponding subpages. The decrypted plaintext after ten rounds is shown at the lower right corner of the `Mix Columns` subpage (Figure 10). The `Substitute Bytes` and `Add Round Key` subpages are the same as in the `Encryption` subpage.

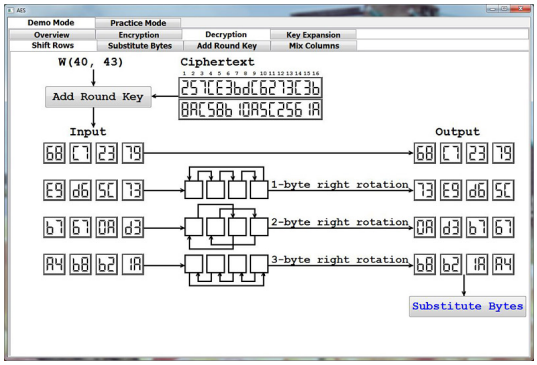

Figure 9: Shift Rows of `Decryption`

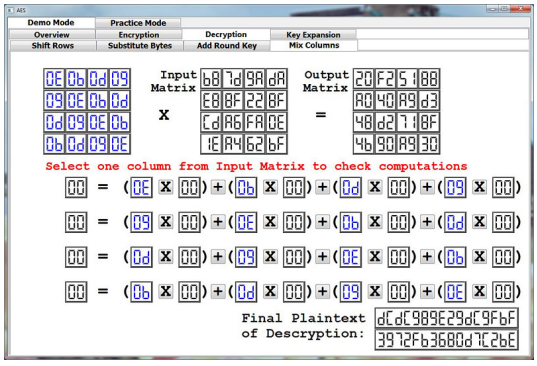

Figure 10: Mix Columns of `Decryption`

3.1.3 *Key Expansion*

This subpage demonstrates how the original 128-bit key is expanded to 44 32-bit words (Figure 11). These words are used in the ten rounds (four words per round) and one initial step for both encryption and decryption (Figure 1). The first four words (in black) are directly derived from the input and all other words (in red) are generated from them. The user may right drag the mouse to move words back and

forth horizontally and click a single word (in blue) to check the word generation procedure. The lower portion of this subpage shows how the four output words with the selected word in blue are obtained using the four input words. Clicking the `G` button brings up the `OperationG` window (Figure 12). The user may also check the XOR (⊕) operations using the `XOR` buttons (Figure 13).

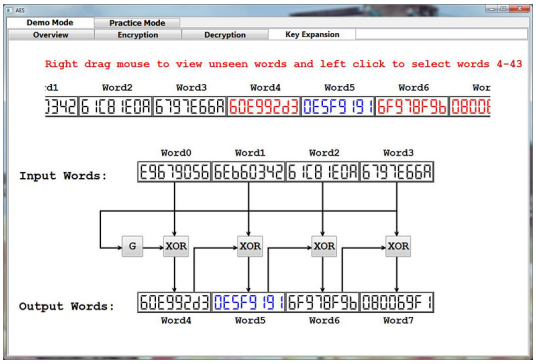

Figure 11: The `Key Expansion` Subpage

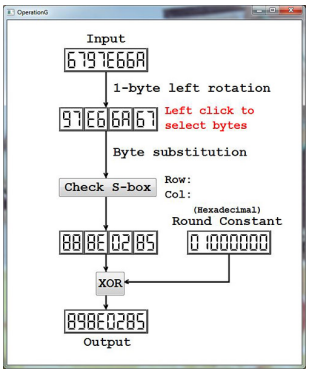

Figure 12: The `Operation G` Window

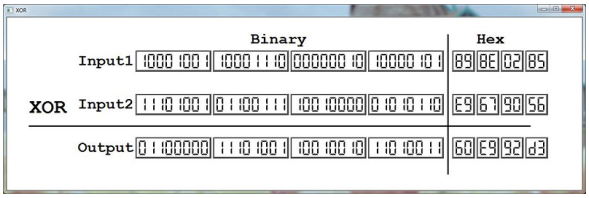

Figure 13: The `XOR` Window

3.2 The Practice Mode

The `Practice` mode follows the same structure of the `Demo` mode (Figure 14), but only supports encryption. The user may step through each computation step; however, all results are hidden and a correct answer is required to advance to the next. The user may click the `Start` button to start a new session by generating a new plaintext-key pair. A dialogue window will pop up to briefly describe the current question and ask the user to enter the answer. The user clicks the `Check Ans` button to verify if the answer is correct and may enter a new answer if the current one is wrong. The `Show Ans` button is provided to show the correct answer and let the user skip the current question. A simple hexadecimal-binary converter is also provided. After the user finishes all questions, a `Completion Report` window will pop up to show the answer to each question with `Correct`, `Wrong` or `Show Ans` if the answer was correct, incorrect or skipped. This report may be sent to the instructor to check the student's completion rate and evaluate the learning effectiveness.

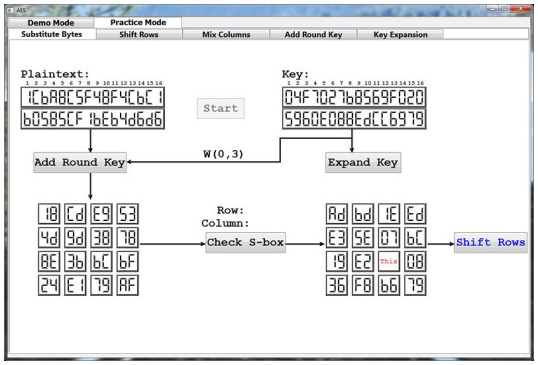

Figure 14: `Practice` Mode of AESvisual

4. EVALUATION AND ASSESSMENT

Our survey consists of two parts: a set of 12 questions and 11 write-in comments. Choices available are 5:strongly agree, 4:agree, 3:neutral, 2:disagree, and 1:strongly disagree. We collected 23 valid forms. The distribution of majors was as follows: 1 in computer network and system administration (CNSA), 8 in electrical and computer engineering (EECE), 9 in computer science, 2 in mathematics, 1 in chemical engineering, and 2 undeclared.

4.1 General Discussion

This paper uses $\alpha = 0.05$ as the level of significance for all statistical decisions. Our survey shows that the students used AESvisual 2.6 times on average during the period of evaluation and the average time they spent on the software was 34.3 minutes with standard deviation and confidence interval 18.9 and $(26.4, 41.0)$.

Table 1: Survey Questions

Q1	The `Demo` mode helped better understand encryption workflow
Q2	The `Demo` mode helped better understand decryption workflow
Q3	The `Demo` mode was helpful for self-study
Q4	The "Mix Columns" module helped understand multiplication and addition in GF(2^8)
Q5	The `Practice` mode helped remember how to encrypt and decrypt
Q6	AESvisual helped identify the parts of AES that I did not understand
Q7	AESvisual helped better understand AES
Q8	AESvisual enhanced the course
Q9	Is AESvisual easy to use?

A summary of the remaining questions is in Table 1. The first three questions Q1, Q2 and Q3 received means 4.04, 4.09 and 3.83, standard deviations 0.64, 0.67 and 0.98, and confidence intervals $(3.79, 4.30)$, $(3.82, 4.35)$ and $(3.45, 4.20)$. This suggested that AESvisual helped students better learn the encryption and decryption flow and for self-study. On the other hand, the `Practice` mode (Q5) was rated slightly

lower with mean, standard deviation and confidence interval 3.70, 0.97 and $(3.31, 4.09)$. The Mix Columns module (Q4) rating was low with mean, standard deviation and confidence interval 3.26, 1.32 and $(2.73, 3.79)$. The next three questions Q6, Q7 and Q8 received good ratings with means 3.87, 3.91 and 3.78, standard deviations 0.97, 0.79 and 0.90, and confidence intervals $(3.48, 4.26)$, $(3.60, 4.23)$ and $(3.42, 4.23)$. This indicated that AESvisual helped student better understand the AES algorithm and that AESvisual did enhance the course. Finally, the easy to use question Q9 was rated with mean 3.48, standard deviation 0.95 and confidence interval $(3.10, 3.86)$.

The Mix Columns component requires the students to have a deeper understanding of $GF(2^8)$ arithmetic to completely comprehend the workflow. This may not be very easy for some students. On the other hand, the low rating of Q4 (3.26) also indicated that the design of AESvisual and the way of presenting the materials require some improvement to be more effective. A few students were not satisfied with the diagram-based design and preferred to have an algorithmic view. In our opinion, the complexity of the $GF(2^8)$ arithmetic and many subpages/steps could have introduced some issues for the students to rate the "easy to use" question (Q9) lower at 3.48. However, the remaining ratings were reasonably high, especially for the `Demo` mode.

4.2 Further Statistical Analysis

The ratings of questions are loosely related to each other. The correlation between every pair of questions was positive. The lowest correlation was 0.18 between Q8 and Q9, which indicated "whether AESvisual enhanced the course" is mostly independent of "whether AESvisual is easy to use". The highest correlation was 0.77 between Q3 and Q7, which suggested that the helpfulness of the `Demo` mode for self-study and the helpfulness of AESvisual to better understand AES were closely related. The correlation between Q1 and Q2 was 0.63, indicating the ratings for the `Demo` mode to better understand encryption workflow and decryption workflow were moderately related to each other.

We also investigated the reaction from different disciplines. We grouped students into three groups: computer science (CS), electrical and computer engineering (EECE), and students from other departments (non-CS). Since the questions may correlate with each other, the questions were also grouped into three groups: (1) Q1, Q2, Q3: the `Demo` mode was helpful, (2) Q6, Q7, Q8: AESvisual was helpful, and (3) all other questions in a single group. We applied MANOVA (multivariate ANOVA) to study the differences among the three student groups on each of the three questions groups. We also applied ANOVA to investigate the difference among all three student groups on each single question.

We used the general linear model (GLM) of R to perform all tests. The p-values for the three groups were 0.72, 0.75 and 0.87. This indicated that the ratings from students in different groups did not vary significantly. The ANOVA result on each single question did not suggest any significant difference either, with the smallest p-value being 0.45 for Q7. In addition, we investigated the difference between CS and EECE using MANOVA on the same question groups and ANOVA on each question. The p-values for the three groups were 0.49, 0.31 and 0.78, indicating that the ratings from CS and EECE did not vary significantly. We did not find significant difference on any single question either. The smallest p-value from the ANOVA results was 0.21 for Q7.

4.3 A Test Score Comparison

A quiz of six problems that address all aspect of the AES cipher was given after the classroom lecture. Then, we discussed AESvisual and made the software available. One week later a second quiz was given. The quiz problems were similar to those of the first. The problems covered Substitute Bytes, Shift Rows, Mix Columns, Add Round Key and Key Expansion. Both quizzes had a full score of 6 points (*i.e.*, one point per problem). We collected 37 and 36 papers from the first quiz and second quiz, and the results are shown in Table 2. The t-values of comparing the means obtained in various t-tests were all larger than 3 with p-values around 0.003, and Cohen's d is 0.73. Thus, the difference between the means is significant and the effect size is reasonably large. As a result, we concluded that the software contributed to student learning significantly.

Table 2: Test Scores

	Quiz 1	*Quiz 2*
Mean	3.32	4.17
St. Dev	1.23	1.13
CI	$(2.93, 3.72)$	$(3.80, 4.54)$

4.4 Student Comments

There are 11 write-in questions asking students to make suggestions for further development. We focused on the following issues: whether only doing the first round of the AES algorithm would be sufficient, whether the Substitute Bytes, Shift Rows, Mix Columns, Add Round Key and Key Expansion modules are helpful, the usefulness of the `Practice` mode, whether the `Demo` mode is more useful than blackboard work, whether new features should be added, and software installation issues.

Students uniformly indicated that only doing the first round of the AES algorithm is sufficient. Of the five modules, only the Mix Columns module received some negative comments. Students indicated that the Substitute Bytes, Shift Rows, Add Round Key and Key Expansion modules were straightforward. Typical comments were "*It [Substitute Bytes] was explanatory and did enhance my learning*", "*The diagrams [of Shift Rows] made it very easy to learn*", "*Not really helpful, rather straight forward to shift rows*", "*It [Add Round Key] did not enhance my learning as much as other modules but it was still helpful*", "*This part was hard for me to figure out until I used the simulation*", and "*This section greatly enhanced my learning by visually showing the full key expansion procedure and operation*".

The Mix Column module was rated the lowest at 3.26. Thus, student comments may provide more information of the possible problems. In general, students felt that the Mix Columns component is the most difficult part of the AES algorithm. Reactions were mixed. Typical positive comments were: "*Helped me understand what I was doing wrong the first time I did the assignment*", "*It made matrix multiplication easier to grasp*", "*The actual process is hard to understand but the tool helped break down the steps and was very helpful to learning*", "*The Mix Columns module does a great job demonstrating the operation*", and "*Allowing the user to select individual columns and see how the output was calculated is very helpful*". Typical negative comments were "*The multiplication steps are still complicated*" and "*This is really the only hard part of AES, and the program did not*

help. (Neither the book nor the program explain multiplication in GF(2^8) field.)". In general, those who provided negative comments indicated that AESvisual did not help step through and did not explain the multiplication and addition over GF(2^8). The textbook [7] explains GF(2^8) arithmetic with polynomials and provids several examples step-by-step. Some students perhaps expected AESvisual to follow these steps closely. We will look into this issue and provide improvements in the future.

Some students believed that the `Demo` mode would be sufficient and they did not use the `Practice` mode. The following has some typical comments: "*I thought the Practice mode was a little unnecessary because the normal mode did a good enough job visually explaining AES*", "*I think it is a useful way for some people to visualize it, but I don't learn that way*", "*Pretty great. It has a nice step-by-step implementation*" and "*I enjoyed it. It made studying easy*".

As for the question "if the `Demo` version helped the students follow the AES algorithm better than the use of the blackboard", most students believed it is useful with typical comments like "*The demo version is quicker than the blackboard and is more organized*", "*The most effective is the step-by-step action of the software. It allowed me to follow along better*", "*I think it did because I learn better visually, which is what this tool provided. Watching values change instantaneously helped*", and "*The Demo mode version did help me more than the use of the blackboard*". On the other hand, a few students suggested that the use of blackboard would help them take notes: "*The blackboard is more helpful to me. It is easier to take notes that way*", "*It helped, but being told about how it works and writing it out helped equally*", and "*I feel you couldn't have one without the other. A basic intro is needed before demoing the software*".

Students did not offer many suggestions for new features. The most needed one was allowing to use user input in the `Demo` mode and in some modules. One student disliked the 7-segment-display font, another suggested to add binary and decimal base notation options, and yet another would prefer to have a web-based version. No significant installation issues were reported.

4.5 Self-Study Investigation

We invited students who did not take our course for a self-study. Since the sample size was small and only three students finished all evaluation forms, no statistical analysis was conducted. This investigation had two stages, each stage took one week. In Stage 1, volunteers were asked to find resources to learn the AES, and at the end they were required to evaluate their progress and complete six quiz problems. In Stage 2, students were provided with AESvisual, and at the end they filled in an evaluation form and completed another six problems. The Stage 2 evaluation form and all quiz problems were identical to those used in class.

Volunteers were usually highly motivated, and, as a result, students received nearly perfect scores in both quizzes. Stage 1 evaluation indicated that the Mix Columns is the most difficult part to understand. Other components are usually considered being straightforward. As a result, they did not have problems in using AESvisual except for the Mix Columns module. However, they did feel that the use of AESvisual was helpful although they still believed that the GF(2^8) arithmetic presentation requires improvement.

Suggestions for further development were not very different from those classroom ones, namely: resizable windows, more colors to distinguish different items, and more pop-up hint windows for explanation and simple exercises.

5. CONCLUSIONS

This paper presented a visualization tool AESvisual for teaching and learning the AES cipher. With this tool, instructors are able to present all the details of AES encryption and decryption, and all complex computation steps, including GF(2^8) addition and multiplication. The `Demo` mode helps students see the flow of the cipher and learn the concepts, and `Practice` mode offers the students an environment to practice the AES encryption. Evaluation results showed that AESvisual was effective in the classroom presentation and for student self-study.

Based on the student comments, the most needed extensions are **(1)** resizable windows, **(2)** allowing the user to enter his input, **(3)** making decimal and hexadecimal input and output possible, **(4)** a better organized and clearer view of the G(2^8) addition and multiplication with explanations, and **(5)** developing a web-based version so that the system would be more "portable" as suggested by some students.

AESvisual is a part of larger development of cryptography visualization tools supported by the National Science Foundation. In addition to AESvisual, VIGvisual for the Vigenère cipher, DESvisual for the DES cipher, RSAvisual for RSA cipher, ECvisual for the elliptic curve based ciphers, and SHAvisual for the Secure Hash Algorithm are available. We hope to complete this development with Diffie-Hellman key exchange, discrete logarithm and digital signature. Tools, evaluation forms, and installation and user guides for Linux, MacOS and Windows can be found at the following link:

`www.cs.mtu.edu/~shene/NSF-4`.

6. REFERENCES

[1] Cryptool. `http://www.cryptool.org`.

[2] O.-S. Chok and S. Herath. Computer Security Learning Laboratory: Implementation of DES and AES Algorithms using Spreadsheets. In *Proceedings of the 37th Midwest Instruction and Computing Symposium*, 2004.

[3] J. Daemen and V. Rijmen. *The Design of Rijndael*. Springer, 2002.

[4] A. McAndrew. Teaching Cryptography with Open-Source Software. In *Proceedings of the 39th ACM SIGCSE Technical Symposium on Computer Science Education*, pages 325–329, 2008.

[5] M. I. Soliman and G. Y. Abozaid. Hardware Visualization of the Advanced Encryption Standard (AES) Algorithm. In *Proceedings of the 18th International Conference on Computer Theory and Applications*, pages 85–93, 2008.

[6] W. Stallings. *Cryptography and Network Security*. Prentice-Hall, third edition, 2003.

[7] W. Trappe and L. C. Washington. *Introduction to Cryptography with Code Theory*. Prentice-Hall, 2002.

Acknowledgment

The authors are supported by the National Science Foundation under grants DUE-1140512, DUE-1245310 and IIS-1319363.