DOUBLE COMPRESSION BASED LOW BANDWIDTH VIDEO TRANSMITTER

Begmatov Shohruh
Assistant teacher, Tashkent University of Information Technologies, Uzbekistan
bek.shohruh@gmail.com

ABSTRACT:

It is very difficult to imagine our modern life without data. As new technologies and data volume increase, sending them quickly is becoming one of the most important issues. Compression methods are commonly used to transfer data faster when transmitting data. Image compression is one of the data compression types used to compress digital images, to reduce their cost for storage or transmission. In this paper, we propose double compression based low bandwidth video transmitter. The device is capable of transmitting data two-stage compression. The first stage is compression based on the existing interpolation method. In the second stage, it is compressed based on the JPEG method. The relationship between bandwidth and quality in image transmission is also analyzed.

Keywords: compression; low bandwith; image interpolation; bitrate.

INTRODUCTION:

In this technological advancement age, one cannot imagine modern life is without digital images. Compression is useful because it reduces the resources required to store and transfer data. Compression and decompression processes consume computing resources. Image interpolation is a widely used method for image resizing and compression. In recent decades, a number of important studies have been conducted on image compression methods, algorithms, and systems. In [1], it is introduced the fundamental theory of image compression, and also described two typical standards - JPEG and JPEG 2000. Lossy techniques such as Quantization, Transform coding, Block Transform Coding; Lossless techniques such as Run Length Coding, Lossless Predictive Coding, Multi-resolution Coding are analyzed in [2]. It is concerned with lossless image compression in [3]. They proposed an approach that is a mix of a number of already existing techniques. Their approach works are based on two stages. first stage, it is applied the well-known Lempel-Ziv-Welch (LZW) algorithm to the image in hand. What comes out of the first step is forward to the second step where the Bose, Chaudhuri and Hocquenghem (BCH) error correction and detected algorithm is used [3]. It is compared different image compression techniques in [4]. It is proposed Discrete Cosine Transform (DCT) based JPEG method in [5]. According to the implementation results of the DCT based JPEG method obtained a higher PSNR value. It is proposed a method for encoding still images based on the JPEG that allows the compression/decompression time cost and image quality to be adjusted to the needs of each application and to the bandwidth conditions of the network [6]. The results presented in [7] show that the DCT exploits interpixel redundancies to render excellent decorrelation for most natural images. It is proposed a novel approach of image compression based on a new method that has been created for image compression which is called the Five Modulus Method (FMM) in [8]. It is proposed a hybrid predictive technique for lossless image compression in [9]. The proposed algorithm achieved by combining predictive Differential Pulse Code Modulation (DPCM) and Integer Wavelet Transform (IWT).

Another interesting study found in [10]. In this study [10], it is coordinated and employed image resizing techniques to replace the widely applied image compression techniques defined by the JPEG. It is presented a comparative analysis of image compression is done by Hybrid (DWT-DCT) Transform in [11]. In [12], is pointed out different basic image compression techniques. Two types of compression methods namely lossy and lossless techniques are analyzed. It is developed an adaptive pattern-based image compression for ultra-low bandwidth weapon seeker image communication in [13]. It is proposed a differential pulse-code modulation based gradient-oriented quantization as the lossy compression algorithm for embedded computer vision systems in [14]. It is proposed a lossy image compression algorithm called Microshift in [15]. It is also proposed a hardware architecture and implement the algorithm on an FPGA. It is presented low bandwidth video-chat compression using deep generative models in [16]. It is proposed an image compression using using discrete cosine transforms and JPEG enconder in [17]. Finally, it is presented compression of video data for transmission over low bandwidth network in [18].

Overall, the overview of the previous contribution on image compression witnesses the tremendous attention devoted during the past decades to the development of various methods, algorithms, and systems. However, none of the aforementioned contributions ([1] - [18]) is likely to develop two stage compression based low bandwidth video transmitter. The proposed video transmitter first compresses the images in two stages based on methods such as interpolation and JPEG, and then transmits the compressed images.

MAIN PART:

Proposed low bandwidth video transmitter is based on double compression. Compression is performed primarily on the basis of the interpolation method. It is then compressed a second time based on JPEG. The main purpose of two-stage compression is to transfer the image quickly by reducing its size. Two-stage compression based low bandwidth video transmitter is presented in Figure 1.

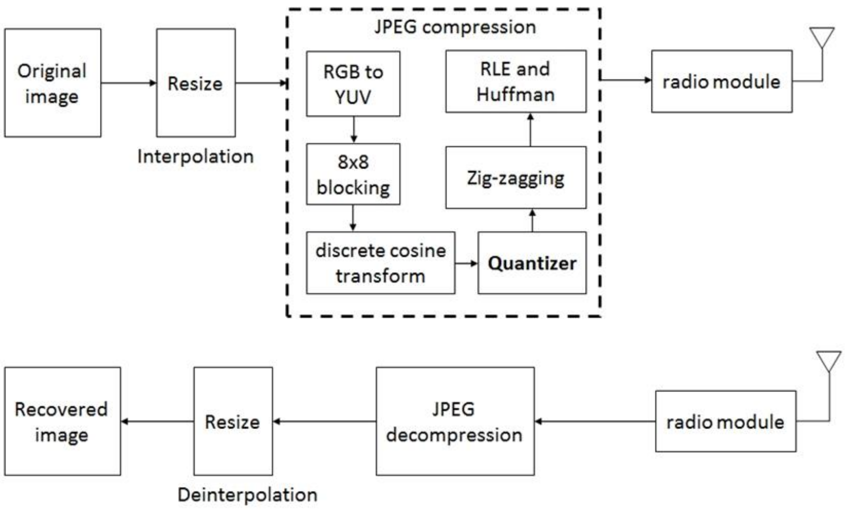
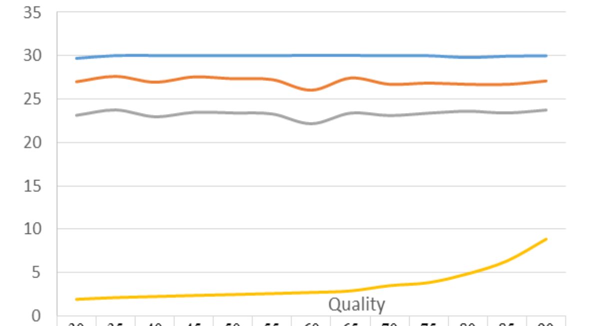


Figure 1. Framework of two-stage compression based low bandwidth video transmitter

We also study on the effect of quality on the bitrate. The relationship between quality and the bitrate is presented in Figure 2.



Quality	30	35	40	45	50	55	60	65	70	75	80	85	90
FPS	29,7	30	30	30	30	30	30,1	30	30	30	29,8	29,9	30,0
Memory(MB)	27	27,6	27	27,6	27,4	27,3	26	27,4	26,7	26,9	26,7	26,6	27,0
CPU	23,1	23,8	23	23,5	23,4	23,3	22,2	23,4	23,1	23,4	23,6	23,4	23,7
Bitrate (Mbps/s)	1,91	2,11	2,23	2,35	2,46	2,57	2,7	2,88	3,49	3,84	4,86	6,34	8,85

Figure 2. Study on the effect of quality on bitrate.

According to the Figure 2, the x-axis represents quality and the y-axis represents FPS, memory, CPU, bitrate. The x-axis is numbered from 30 to 90 and the y-axis is numbered from 0 to 35. The four parameters studied were described in the figure 2 in color as follows: blue for FPS; brown for memory; gray for CPU; and yellow for bitrate. When the quality level was 30, FPS showed 29.7, memory usage (RAM) 27 MB, CPU usage 23.13%, and bitrate 1.91 Mbps. When the quality level was 60, FPS showed 30.05, memory usage (RAM) was 26.03 MB, CPU usage was 22.16%, and bitrate was 2.7 Mbps. When the quality level was 90, FPS 30 showed that memory usage (RAM) was 27.09 MB, CPU usage was 23.74%, and bitrate was 8,85 Mbps. Thus, it can be seen that the higher the quality, the higher the bitrate, according to the four parameters studied in the dependence of quality on the bitrate.

ANALYSIS OF INTERPOLATION METHODS

In this section, several types of interpolation method used in image compression are discussed and compared. The nearest neighbor, bicubic, bilinear and area methods of interpolation methods are analyzed and the most effective of them (for low bitrate) is chosen.

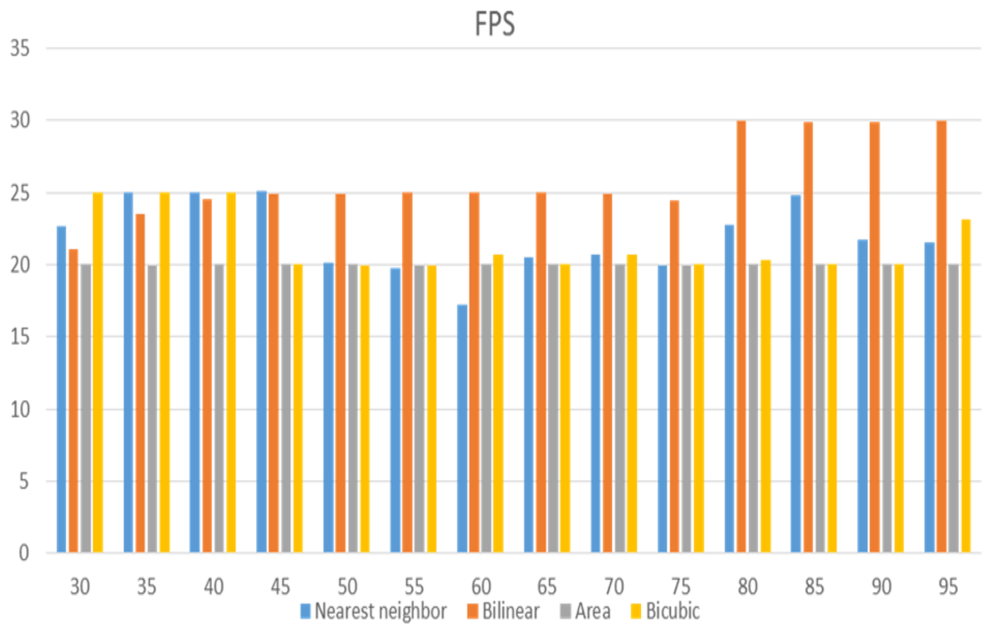


Figure 3. Comparison of the interpolation methods through FPS and quality

As can be seen from the Figure 3, the x axis represents quality and the y axis represents FPS. The x-axis is numbered from 30 to 95 and the y-axis is numbered from 0 to 35. The four methods studied were described in the figure in color as follows: blue for Nearest neighbor method; brown for Bilinear method; gray for Area method; and yellow for Bicubic method. It turns out that the FPS can go up to 25 when the quality level is 30. This means that the Bicubic method is superior to the other three methods at this level of quality. When the quality level was increased to 35 and 40, the results in the Nearest neighbor method and the Bicubic method showed that they were better than the other two methods and could reach 25 FPS. When the quality levels are 50 and 55, the results obtained by the Nearest Neighbor method, Area method and Bicubic method can reach almost the same value, ie 20 FPS, and the result obtained by the Bilinear method is relatively superior and can reach 25 FPS. When the quality level ranged from 60 to 95, the results obtained using the Bilinear method showed that it was much higher than the other three methods and could reach up to 30 FPS. Thus, the Bilinear method turned out to be the best among the methods studied in relation to quality FPS.

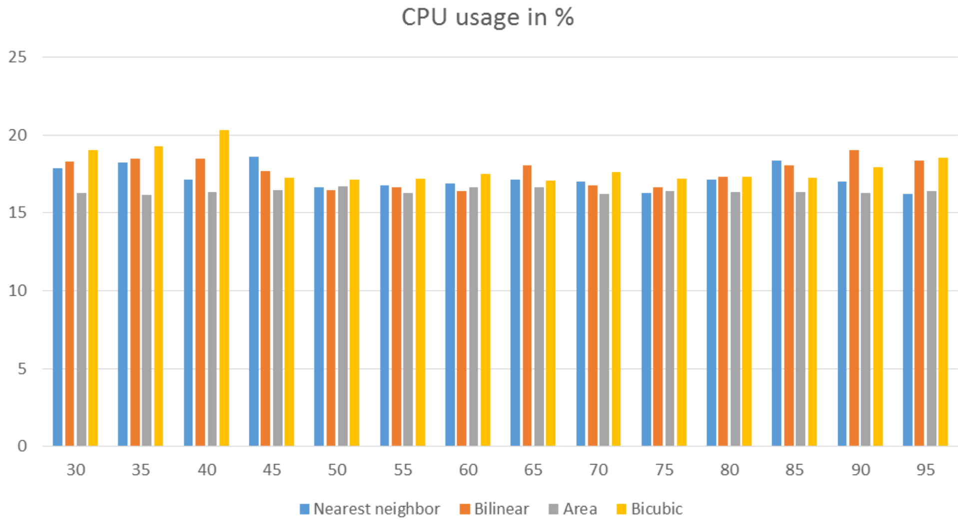


Figure 4. Comparison of the interpolation methods through CPU usage and quality

As can be seen from the Figure 4, the x axis represents quality and the y axis represents CPU usage. The x-axis is numbered from 30 to 95 and the y-axis is numbered from 0 to 25 and expressed as a percentage. The four methods studied were described in the figure 4 in color as follows: blue for Nearest neighbor method; brown for Bilinear method; gray for Area method; and yellow for Bicubic method. When the quality level was 30, the results obtained by the Bilinear method used almost 18% CPU usage, while the lowest, i.e., almost 16% CPU usage, corresponded to the results obtained based on the Area method. Area method even when the quality level is increased to 35 and 40 was the lowest CPU usage method, while Bicubic, by contrast, again had the highest CPU usage of more than 20%. When the quality level was 45, the Nearest neighbor method had the highest CPU usage of almost 19%, while the results based on the Area method had the lowest CPU usage of almost 17%. When the quality level was between 50 and 95, the results obtained based on the relatively more Bicubic method required the most CPU usage, while the results obtained based on the Area method required the least CPU usage in most cases. Thus, the Area method turned out to be the best and the Bicubic method the worst among the methods studied in relation to quality CPU usage.

Comparison of the interpolation methods through RAM usage and quality is presented in Figure 5.

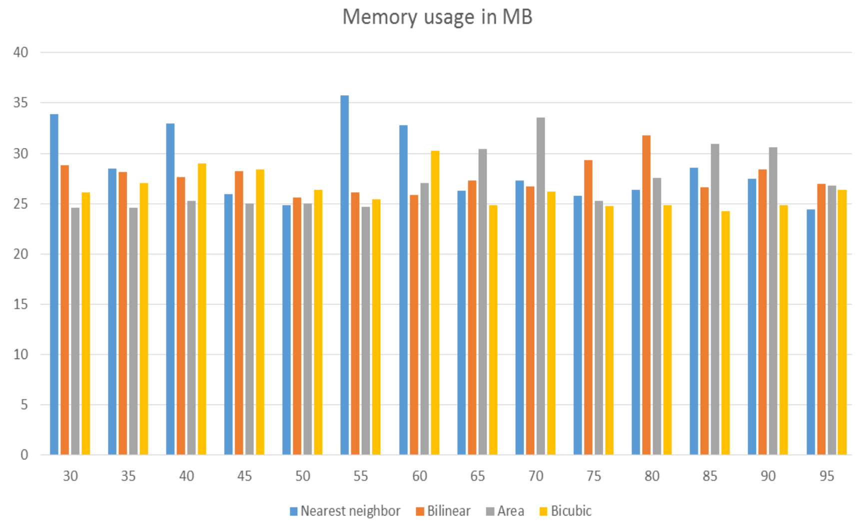


Figure 5. Comparison of the interpolation methods through RAM usage and quality

According to the figure above, the x-axis represents quality and the y-axis represents memory usage, i.e. RAM usage. The x-axis is numbered from 30 to 95 and the y-axis is numbered from 0 to 40 and expressed in MB. The four methods studied were described in the figure 5 in color as follows: blue for Nearest neighbor method; brown for Bilinear method; gray for Area method; and yellow for Bicubic method. When the quality level was 30, the results obtained by the Nearest neighbor method took up the most almost 34 MB of RAM, close to 29 MB according to the Bilinear method, close to 26 MB according to the Bicubic method, and almost the least. The 25 MB RAM usage was based on the Area method. Even when the quality level is increased to 35 and 40, the Nearest neighbor method has the most RAM usage and the Area method achieved the lowest RAM usage. When the quality levels were 45 and 50, the results for the Bicubic method achieved the highest RAM usage, and for the Area method the lowest RAM usage. When the quality level was 55, the results obtained by the Nearest method were significantly larger than the remaining methods, with almost 36 MB of RAM usage, while the rest were close to 26 MB on Bilinear, slightly more than 25 MB on Bicubic, and the lowest Nearly 25 MB by area method. When the quality level was 60, the results obtained by the Nearest method took up the most space in RAM close to 33 MB, and the lowest was the Bilinear method, which was slightly more than 25 MB. When the quality level is 65, the Area method takes up the most RAM, 30 MB, and the Bicubic method takes up the least 25 MB. When the quality level was 70, the Area method was again the method that took up the most RAM, and the Bicubic method was the method that took up the least space. When the quality levels were 75 and 80, the Bilinear method was the method that took up the most RAM, and Bicubic was the method that took up the least space. When the quality levels were 85 and 90, the Area method was the method that took up the most RAM, and Bicubic was the method that took up the least space. When the quality level was 95, the Bilinear method took up the most space from RAM and the Nearest neighbor method took the least space. Thus, the Area method turned out to be the best and the Nearest neighbor method the worst among the methods studied in relation to quality RAM usage.

Comparison of the interpolation methods through bitrate (bandwidth) usage and quality is presented in Figure 6.

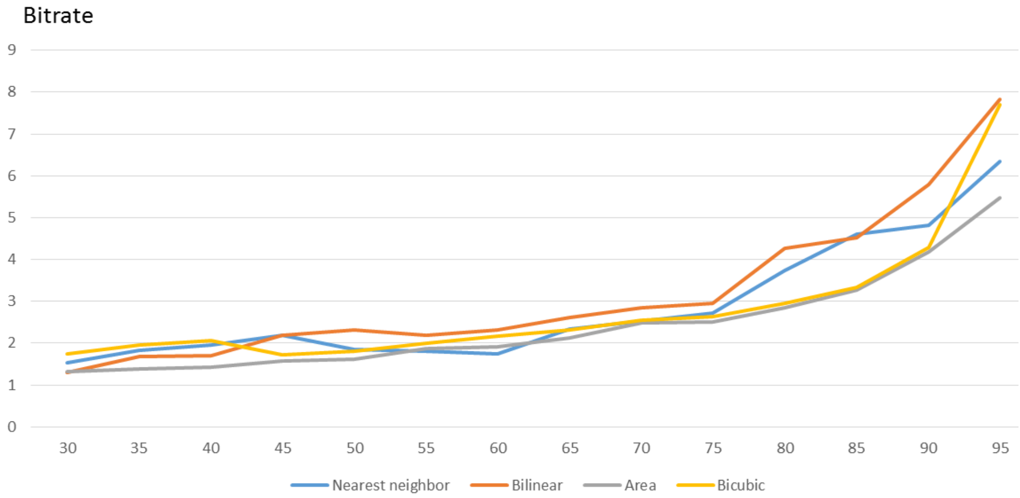


Figure 6. Comparison of the interpolation methods through Bitrate and quality

According to Figure 6, the x-axis represents quality and the y-axis represents bitrate. The x-axis is numbered from 30 to 95 and the y-axis is numbered from 0 to 9 and expressed in bps. The four methods studied were described in the figure 6 in color as follows: blue for Nearest neighbor method; brown for Bilinear method; gray for Area method; and yellow for Bicubic method.

When the quality level was 30, the results obtained by the Bilinear and Area method achieved the same bitrate bitrate, and this was the lowest bitrate. The highest bitrate was observed in the Bicubic method. Even when the quality levels were increased to 35 and 40, the highest bitrate Bicubic and the lowest bitrate Area method were met. When the quality level was 45, the Nearest neighbor and the Bilinear method achieved the same and highest bitrate, respectively, while the lowest bitrate corresponded to the Area method. When the quality level was 50, the results obtained by the Bilinear method reached the highest bitrate, hile the lowest bitrate again corresponded to the Area method.

When the quality level was 55, the results obtained by the Bilinear method reached the highest bitrate, while the lowest bitrate corresponded to the Nearest neighbor method. When the quality level was 60, the results obtained by the Bilinear method again reached the highest bitrate, while the lowest bitrate again corresponded to the Nearest neighbor method.

When the quality level was 65, the results obtained by the Bilinear method reached the highest bitrate, while the lowest bitrate again corresponded to the Area method. When the quality level ranged from 70 to 95, the results obtained by the Bilinear method reached the highest bitrate, while the lowest bitrate again corresponded to the Area method. Thus, the Bilinear method turned out to be the best and the Area method the worst among the four methods studied in terms of quality bitrate (bandwidth) dependence.

CONCLUSION:

In this paper, we propose double compression based low bandwidth video transmitter. The device reduces the amount of data based on two-stage compression and is able to transfer the image that is being compressed. From the results of the research, it can be concluded that the higher the bitrate (bandwidth), the higher the quality.

REFERENCES

[1] W. Wei, "An Introduction to Image Compression," *National Taiwan University, Taipei, Taiwan, ROC*. 2008.

[2] K. Rajandeep and C. Pooja, "A review of image compression techniques," *Int. J. Comput. Appl. (0975 – 8887)*, vol. 142, no. 1, pp. 8–11, 2016.

[3] A. Alarabeyyat *et al.*, "Lossless Image Compression Technique Using Combination Methods," *J. Softw. Eng. Appl.*, vol. 05, no. 10, pp. 752–763, 2012.

[4] S. Somvanshi and S. Rahmani, "Comparing Different Image Compression Techniques," *Int. Res. J. Eng. Technol.*, vol. 7, no. July, pp. 837–839, 2020.

[5] Y. Devi, "JPEG Image Compression Using Various Algorithms : A Review," *Int. J. Comput. Sci. Trends Technol.*, vol. 4, no. 3, pp. 89–92, 2016.

[6] J. M. Pascual, H. M. Mora, A. Fuster Guilló, and J. Azorín López, "Adjustable compression method for still JPEG images," *Signal Process. Image Commun.*, vol. 32, pp. 16–32, 2015.

[7] A. Kaushik and M. Gupta, "Analysis of Image Compression Algorithms," *Int. J. Eng. Res. Appl.*, vol. 2, no. 2, pp. 773–779, 2012.

[8] F. A. Jassim and H. E. Qassim, "Five Modulus Method for Image Compression," *Signal Image Process. An Int. J.*, vol. 3, no. 5, pp. 19–28, 2012.

[9] N. A. N. Azman, S. Ali, R. A. Rashid, F. A. Saparudin, and M. A. Sarijari, "A hybrid predictive technique for lossless image compression," *Bull. Electr. Eng. Informatics*, vol. 8, no. 4, pp. 1289–1296, 2019.

[10] J. J. Shen, C. H. Yeh, and J. K. Jan, "A new approach of lossy image compression based on hybrid image resizing techniques," *Int. Arab J. Inf. Technol.*, vol. 16, no. 2, pp. 226–235, 2019.

[11] G. Singh and J. S. Sandhu, "Comparative analysis of image compression techniques," *Int. J. Eng. Res. Technol.*, vol. 5, no. 5, 2017.

[12] D. V Rojatkar, N. D. Borkar, B. R. Naik, and R. N. Peddiwar, "Image Compression Techniques: Lossy and Lossless," *Int. J. Eng. Res. Gen. Sci.*, vol. 3, no. 2, pp. 912–917, 2015.

[13] H. Wei *et al.*, "Adaptive pattern-based image compression for ultra-low bandwidth weapon seeker image communication," *Vis. Inf. Process. XVIII*, vol. 7341, no. April 2009, p. 73410B, 2009.

[14] L. Guo, D. Zhou, J. Zhou, S. Kimura, and S. Goto, "Lossy Compression for Embedded Computer Vision Systems," *IEEE Access*, vol. 6, pp. 39385–39397, 2018.

[15] B. Zhang, P. V. Sander, C. Y. Tsui, and A. Bermak, "Microshift: An Efficient Image Compression Algorithm for Hardware," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 11, pp. 3430–3443, 2019.

[16] M. Oquab *et al.*, "Low Bandwidth Video-Chat Compression using Deep Generative Models," 2020.

[17] R. Mathey, "An image compression using using discrete cosine transforms and JPEG enconder," *IOSR J. Comput. Eng.*, vol. 16, no. 2, pp. 01–05, 2014.

[18] M. P. G. Dr. Dhirendra Mishra, "Compression of Video Data for Transmission Over Low Bandwidth Network," *Int. J. Eng. Res. Technol.*, vol. 3, no. 12, pp. 102–107, 2014.