Alpha-Gamma Data Compression Method for Artificial Vision Systems using Visual Cortex Stimulation

Tomoki SUGIURA[†1], Arif Ullah Khan[†1], Yoshinori TAKEUCHI[†1] and Masaharu IMAI[†1†2]

[†1] Graduate School of Information Science and Technology, Osaka University
[†2] The Center for Advanced Medical Engineering and Informatics, Osaka University
1-5 Yamadaoka, Suita, Osaka, 565-0871, JAPAN
{s-tomoki,khan,takeuchi,imai}@ist.osaka-u.ac.jp

Abstract– In this paper a data compression method for visual cortex stimulation based artificial vision is proposed and evaluated. The proposed method uses run-length encoding to express visual cortex stimulus data in numerical form, in which the numerical data representing '1' data and '0' data are encoded into binary by alpha encoding and gamma encoding, respectively. From experimental results, the proposed method reduced data size approximately 83% while execution cycles of the proposed method is practically equal to gamma encoding.

I. Introduction

It is common for a person who is physically handicapped to use a cane or a wheelchair for his or her mobility, but there are few treatment techniques for restoring or replacing their sense organs.

In Japan, there are about 300 thousand people to whom physical disability certificates are issued [1]. Among these, more than 15% people are over 65 years old, who are classified into older people [2]. In Japan, vision disorder is mainly caused by glaucoma or macular degeneration due to aging and diabetic retinopathy due to lifestyle disease. Thus, it is expected that the number of people who lost their vision by the aging increases dramatically because Japan enters into an aging society. There is an urgent need for finding solutions for treatment of vision disorder.

One solution for restoring vision is to use artificial vision systems. The systems aim at obtaining pseudo vision by stimulation of visual nerves artificially [3]. They utilize phosphene phenomenon by stimulating optic nerves using electric signals [4]. The systems express visual inputs as a set of dots. The artificial vision systems are classified into several types such as retina stimulation, visual cortex stimulation etc. A project team in Osaka University is developing a prototype for artificial vision system based on visual cortex stimulation. The system uses the silicon retina system proposed by [5] as an input camera device. The system captures visual data from the silicon retina system, transforms the visual data to stimulus signal for visual cortex, and wirelessly transfers the stimulus signal data to multiple electrode array attached to the visual cortex.

One of bottlenecks of the system is power consumption which not only affects mobility and portability due to larger battery size but also causes inconvenience to the user because of extra heat generation. Usually, the power consumption of a wireless communication system is directly proportional to amount of traffic data. Therefore, compressing and reducing data size before communication will reduce not only power consumption of the wireless communication system but also data requirement of bandwidth.

In this paper a new method of data compression is proposed, which compresses data efficiently and calculates in short time, to reduce power consumption of the artificial vision system with the silicon retina system as compared to conventional methods. The α encoding, the γ encoding, and the δ encoding are previously proposed in [6] by Elias. According to the analysis of the data transferred in the artificial vision system, the run-length encoding is used, and usage of the α encoding and the γ encoding has good performance in order to compress its data.

The rest of this paper is organized as follows: Sect. II explains organization of the artificial vision system. Sect. III explains stimulus signal data of the artificial vision system. In Sect. IV, proposed data compression method is introduced. In Sect. V, experimental results are presented, and conclusion and future work are summarized in Sect. VI.

II. The Artificial Vision System

This section introduces stimulus data of the artificial vision system with silicon retina.

A. Overview

This artificial vision system is developed by Osaka University, which is divided into two parts. The one is an outer unit attached to the top of the skull, and the other is an inner unit attached on the visual cortex.

The outer unit is composed of the silicon retina system, a processor for calculation, and a wireless communication unit to exchange data with the inner unit. The silicon retina system can perform high-speed image processing, which mimics

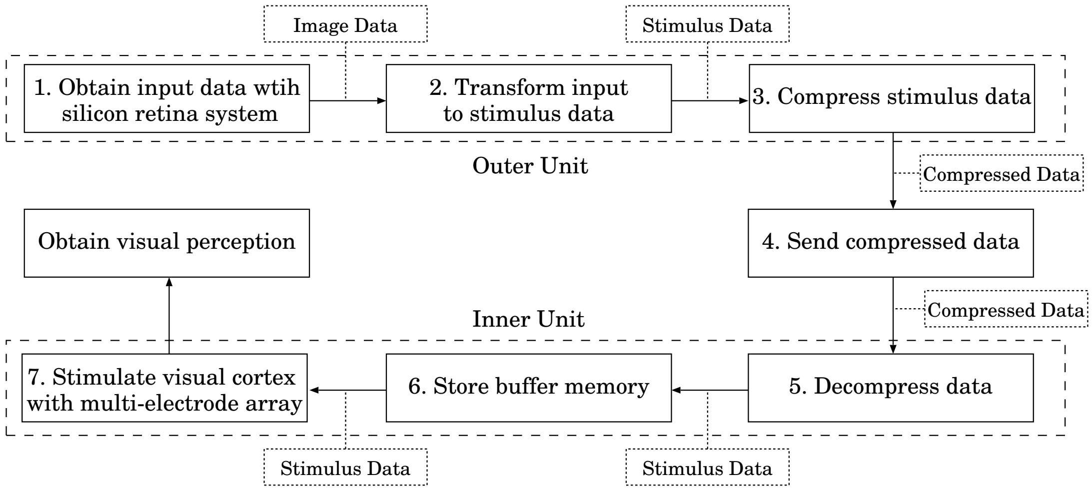


Fig. 1. Organization of the artificial vision system

human retina with analog CMOS integrated circuits [7], and output the stimulus data for the visual cortex. The processor of the outer unit transfers data to the wireless communication unit and compresses stimulus data.

The inner unit is composed of multi-electrode arrays, a stimulus control device to control the arrays, a processor for calculation and control, and a wireless communication unit. When the processor receives compressed data via the wireless communication unit, it decompresses the received data and transfers decoded data to buffer memory in the stimulus control device. Then, it stimulates visual cortex for using multi-electrode arrays based on the data in the buffer memory.

The silicon retina system generates the stimulus data, which consist of stimulation positions, stimulation timings, strength for stimulation by multi-electrode arrays. This paper studies stimulation position data because stimulus position data is most of stimulus data.

B. *The Artificial Vision System*

Fig.1 shows the organization of the artificial vision system and data. The system works in the following steps:

1. The system obtains input image data by the silicon retina system,
2. transforms input image data to stimulus data,
3. compresses the stimulus data,
4. sends compressed data to the inner unit,
5. decompresses the compressed data to the stimulus data,
6. stores the stimulus data in the stimulus control device,
7. and stimulates visual cortex with the multi-electrode array.

This stimulation generates visual perception to users.

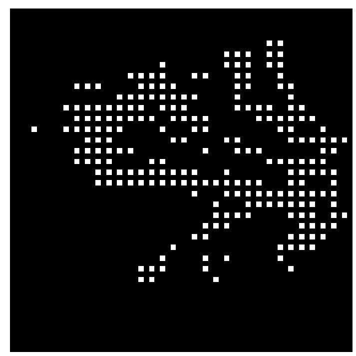


Fig. 2. Example of stimulus data

III. Characteristic of Stimulus Data

The expression of the stimulus data and the statistical analysis of successive bits in stimulus data are explained in this section.

A. *Stimulus Data*

Fig.2 shows an example of an image representing stimulus data. The stimulus data of preliminarily system is composed of 32×32, 1024 bit in all. The data express whether the stimulus by an electrode related with its position of data exists or not. If the data is '1', represented white dot in fig.2, the electrode stimulates the nerve of cortex. If the data is '0', represented black dot in fig.2, the electrode does not.

B. *Analysis of Stimulus Data*

In fig.2, black dots, which equal to '0' bits, occupy much space and most of all these dots are successive in long length seen in the horizontal direction. In other hand, some white

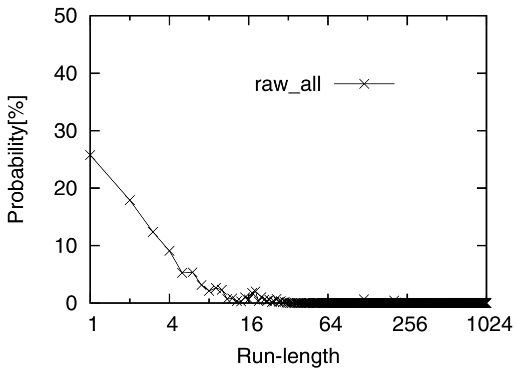


Fig. 3. Probability of all run-length in stimulus data

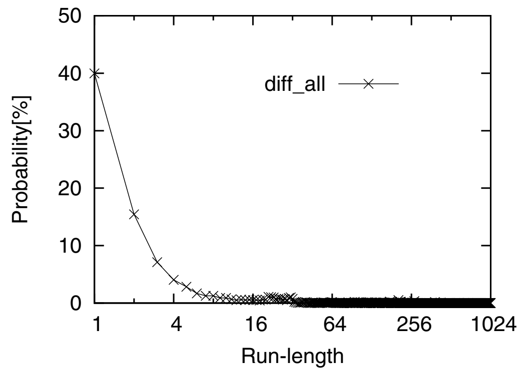


Fig. 6. Probability of all run-length data in difference data

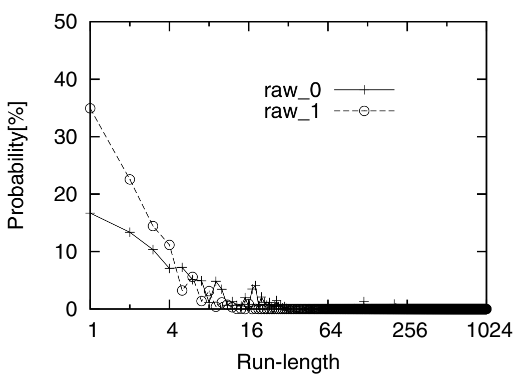


Fig. 4. Probability of 0's and 1's run-length in stimulus data

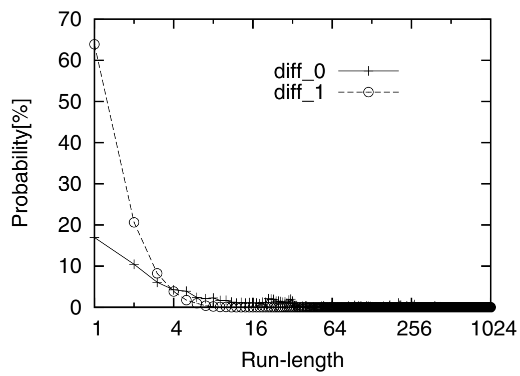


Fig. 7. Probability of 0's and 1's run-length data in difference data

dots, which equal to '1' bits, are isolated and others are successive in short length. As mentioned above, there is high possibility that the run-length encoding is efficient for representation of the stimulus data in numerical forms.

The data analysis is conducted data from the silicon retina system. Objects of the data consists of walking people, running motorcars and motor bicycles on the road. 13 samples of scenes are filmed in various situation such as on the narrow roads, in the cross of arterial roads, and at the station.

Fig.3 shows the probability of run-length of a sample which consists of 273 frames as "raw_all". Fig.4 shows the probability of 0's and 1's run-length of the same sample as "raw_0" and "raw_1" respectively. The x-axes of fig.3 and fig.4 represent run-length, and the y-axes represent the probability of data of run-length.

As shown in fig.3, run-length data tend to be short. In stimulus data, run-length data whose length is one is dominant and occupies about 25%, and run-length data that are less than and equal to 16 are occupy about 90%. Fig.4 shows that the distribution of probability of 0's run-length data is different from that of 1's run-length data. In other words, 1's run-length is shorter than 0's run-length. Run-length data whose length is one occupies about 35% in 1's run-length, while it occupies about 16% in 0's run-length data. Furthermore, run-length data whose length are less than and equal to 8 occupy about 96% in 1's run-length data, but on the contrary, they occupy about 65% in 0's run-length data.

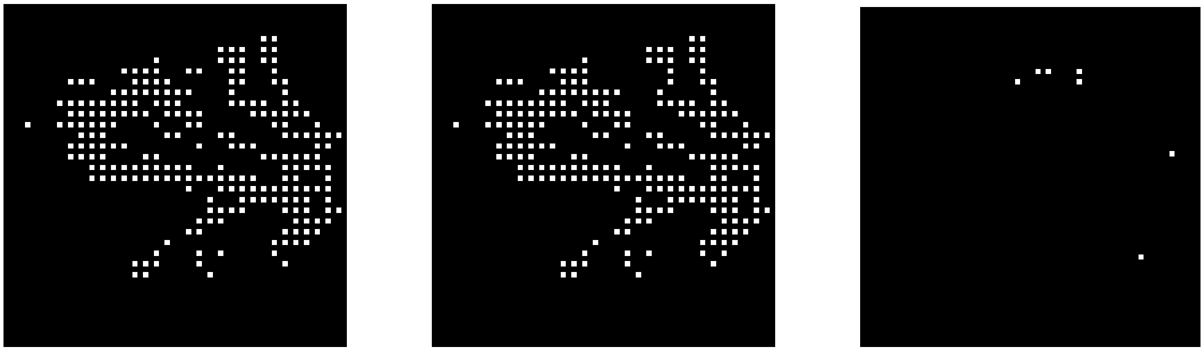
C. Difference of Successive Stimulus Data

Fig.5(a) and fig.5(b) are stimulus data and they are continuous frame in a sample data. It is clear that they are extremely similar. Fig.5(c) shows the difference of fig.5(a) and fig.5(b).

As the data in fig.5(c) indicates, there are more successive black dots and less isolated white dots in the difference data than stimulus data shown as fig.5(a) and fig.5(b). Stimulus data is called raw data in distinction from difference data as follows in this paper. Distribution of run-length data in the difference data is analyzed, and the same data in Sect.III-B is used in this analysis.

Fig.6 shows the probability of run-length data of difference data as "diff_all". Fig.7 shows the probability of 0's and 1's run-length data of the same sample, which is presented as "diff_0" and "diff_1", respectively. The x-axes of fig.6 and fig.7 represents run-length and the y-axes represents the rate of probability of run-length data in the sample.

Compared fig.6 with fig.3, the distribution of run-length in the difference data is shorter than that in the raw data. In the difference data, run-length data whose length is one is domi-



(a) Previous frame of stimulus data (b) Current frame of stimulus data (c) Difference of current and previous stimulus data

Fig. 5. Example of difference data

nant and occupies about 40%, about 25% in the raw data. On the other hand, run-length data that are less than and equal to 16 occupy about 79%, about 90% in the raw data. As a clear difference, run-length data whose length are more than 32 occupy about 7% in the difference data, on the contrary, about 1% in the raw data. In addition, fig.7 shows that the 1's run-length data is extremely biased to be short in the difference data. In 1's run-length data in the difference data, data whose length is one occupies about 64%, which is nearly twice as much as that in the raw data. In 0's run-length data in the difference data, data whose length are less and equal to 8 occupy about 40%, which is not similar to the data in the raw data about 65%. 0's run-length data whose length are more than 32 are occupy about 20% in the difference data while they occupy about 3% in the raw data.

From the above analysis, there is different distribution between 0's run-length data and 1's run-length data in the stimulus data. Furthermore, it is effective for the run-length data to trim distribution of that with using difference data generated from two continuous stimulus data.

IV. Proposed Encoding Method

This section introduces the proposed method for compression for stimulus data.

A. Encoding for 0's Run-length Data

The proposed method uses the γ encoding or the δ encoding to encode 0's run-length data. Table I shows a few examples of γ and δ code words for integers. In the γ encoding, the number of leading zero as the header of code words represents the width of encoded data and the number of leading zero is equal to $\lfloor \log_2 n \rfloor$. Code word $C_\gamma(n)$ is a concatenation of the header and expression of n in binary, therefore the length of $C_\gamma(n)$ is as follows:

$$L[C_\gamma(n)] = 2\lfloor \log_2 n \rfloor + 1, \tag{1}$$

where L indicates the length of codes.

TABLE I
Code Words of γ and δ Encoding

n	$C_\gamma(n)$	$C_\delta(n)$
1	1	1
2	010	0100
3	011	0101
4	00100	01100
5	00101	01101
6	00110	01110
7	00111	01111
8	0001000	00100000
⋮	⋮	⋮
15	0001111	00100111
16	000010000	00101000
⋮	⋮	⋮

In the γ encoding, the width of encoded data is encoded by γ encoding as the header of code word. Code word $C_\delta(n)$ is a concatenation of the header and expression of n in binary, which omits the most significant bit. Therefore, the length of $C_\delta(n)$ is as follows:

$$L[C_\delta(n)] = 2\lfloor(\lfloor \log_2 n \rfloor + 1)\rfloor + \lfloor \log_2 n \rfloor + 1. \tag{2}$$

The γ encoding and δ encoding do not require storing its record of encoding such as Huffman coding, so that it is expected that the encoding process takes less time and less amount of data size. γ encoding is superior to δ encoding in the case of encoding small numerical number up to 32. Therefore, evaluation of γ encoding and δ encoding are conducted in the following section.

B. Encoding for 1's Run-length Data

The proposed method uses the α encoding to encode 1's run-length. Table II shows a few examples of α code words of

TABLE II

CODE WORDS OF α ENCODING

n	$C_\alpha(n)$
1	1
2	01
3	001
4	0001
5	00001
6	000001
7	0000001
8	00000001
$\vdots$	$\vdots$

integer. In the α encoding, the number of leading zero as the header of code word is equal to $n-1$. Code word $C_\alpha(n)$ is a concatenation of the header and a single '1' bit. Therefore, the length of $C_\alpha(n)$ is as follows:

$$L[C_\alpha(n)] = n. \tag{3}$$

α encoding is superior to γ encoding and δ encoding in the case of encoding small numerical number up to five from table I and table II. As shown in fig.3 and fig.6, there are almost all short 1's run-length data up to 3 in both the raw data and difference data. Therefore it is expected that using α encoding to compress 1's run-length is more efficient about compression the stimulus data than the other encoding methods.

C. Data Compression Method using α-γ Encoding

To improve the efficiency of encoding, the proposed method encodes the difference data, which is generated by the continuous two stimulus data. Considering the results of the analysis, to use the difference data is expected that the difference data helps the encoding methods used in the proposed method more effective than the raw data.

The method which uses γ encoding for 0's run-length and α encoding for 1's run-length is called α-γ encoding and which use δ encoding for 0's run-length and α encoding for 1's run-length is called α-δ encoding in this paper.

Proposed method compresses stimulus in the following steps:

1. The proposed method calculates the difference data by the continuous two stimulus data,
2. starts at '0' bit on the initial state,
3. encodes 0's run-length by γ encoding,
4. encodes 1's run-length by α encoding,
5. determines the end of the difference data and goes back 3 if the data remains.

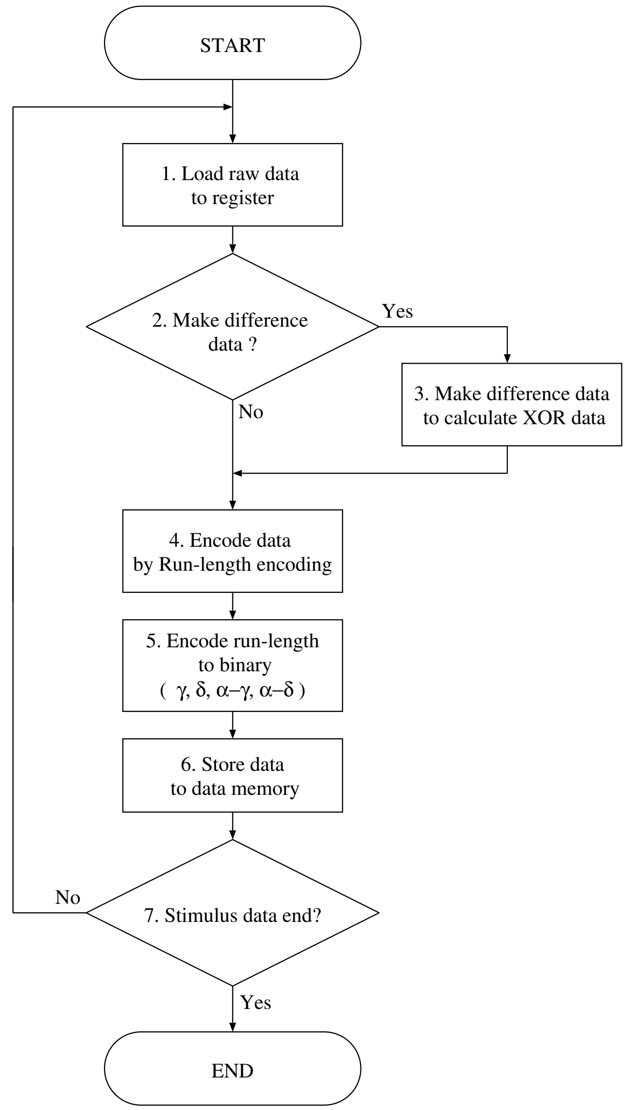


Fig. 8. Flowchart of data compression

V. EXPERIMENTS

This section explains experimental results for evaluating the proposed compression method. Data compression ratio R is defined as follows:

$$R[\%] = 100 \times \left(1 - \frac{D_\mathrm{c}[\mathrm{bit}]}{D_\mathrm{r}[\mathrm{bit}]}\right), \tag{4}$$

where D_c is compressed data size and D_r is raw data size, which is equal to 1024.

A. Experimental Setup

The data compression ratio is compared by encoding the raw data and the difference data. Fig.8 shows the flowchart of data compression. In order to compare the proposed methods, other encoding methods for compression as follows are used: the γ encoding and the δ encoding. The α encoding is not used in

TABLE III
DATA COMPRESSION RATIO [%]

Target	α	γ	$\alpha - \gamma$	δ	$\alpha - \delta$
raw data	0.00	69.84	72.04	68.27	70.34
difference data	0.00	83.28	83.63	83.25	84.03

TABLE IV
EXECUTION CYCLES FOR ENCODE [CYCLE]

Target	γ	$\alpha - \gamma$	δ	$\alpha - \delta$
raw data	35,799	35,518	41,187	37,994
difference data	29,977	29,566	32,236	30,795

the experiments due to its property of encoding. The length of the α encoding's code words is equal to the length of encoded numerical number, so the data size after compression is the same before compression. In short, the compression using only the α encoding is not efficient as regards compression ratio: $R = 0$. The samples of the experiments are gathered by the silicon retina system. Thirteen samples which consists of 8718 frames are used in the experiments.

16bit RISC processor, Brownie Micro 16 provided by ASIP Solutions, Inc., is used for experiments. To measure execution cycles for compression method, ModelSim provided Mentor, Inc. is used to simulate these methods. First, data compression rate over raw data are compared. Second, data compression rate over difference data are compared. Then, execution cycle of compression method is compared.

B. Experimental Results

B.1. Data Compression Ratio

Table III shows comparison on data compression ratio with α encoding, γ encoding, δ encoding, α-δ encoding, and proposed α-γ encoding. The results show that the compression ratios of the difference data are improved compared with those of the raw data. Proposed method achieves more than 83% data compression ratio. The case of compressing the difference data by the α-δ encoding is the maximum compression ratio, and there are slightly increased compression ratio of the α-γ and the α-γ encoding compared from the methods which do not use the α encoding.

B.2. Execution Cycles

Table IV shows comparison on execution cycles with the proposed methods and the previous methods. First, the execution cycles in the case of compressing the difference data are smaller than the raw data. Execution cycles of the difference data compression decrease about 20% by the raw data compression. Execution cycles of the difference data compression in the proposed method decrease about 1.5%, 9.0%, 4.2% compared with the γ encoding, the δ encoding and α-δ encoding, respectively.

VI. CONCLUSIONS AND FUTURE WORK

This paper proposed a compression method, which is called $\alpha - \gamma$ encoding, specified for encoding the stimulus data used by the artificial vision system with the silicon retina system. The statistical analysis of the stimulus data shows the data has different distribution between 0's run-length and 1's run-length. Predicated on its result, the proposed method makes the difference data to calculate the difference data of two continuous stimulus data and compresses 0's run-length and 1's run-length by α encoding and γ encoding, respectively. The proposed method increased compression ratio and reduced execution cycles for encoding.

Future work includes consideration of total system including compression processing and implementation method of data compression and decompression.

ACKNOWLEDGEMENT

This work was partly supported by Regional Innovation Strategy Support Program of the Ministry of Education, Culture, Sports, Science and Technology in Japan (MEXT), and the Center for Advanced Medical Engineering and Informatics (MEI center) of Osaka University. We express our special thanks to Prof. Tetsuya Yagi, Prof. Seiji Kameda from Osaka University for their valuable discussions about artificial vision systems.

REFERENCES

[1] C. Office, "Annual report on government measures for persons with disabilities(summary) 2012," http://www8.cao.go.jp/shougai/english/annualreport/2012/index-pdf.html.

[2] L. Ministry of Health and Welfare, "Health care and welfare measures for people with physical disabilities," 2011, http://www.mhlw.go.jp/english/policy/care-welfare/welfare-disabilities/index.html.

[3] W. H. Dobelle, M. G. Mladejovsky, and J. P. Girvin, "Artificial vision for the blind: Electrical stimulation of visual cortex offers hope for a functional prosthesis." *Science*, Vol. 183, No. 4123, pp. 440–444, 1974.

[4] A. M. Potts and J. Inoue, "The electrically evoked response of the visual system (EER): III. further contribution to the origin of the EER," *Investigative Ophthalmology & Visual Science*, Vol. 9, No. 10, pp. 814–819, 1970.

[5] H. Okuno and T. Yagi, "Image sensor system with bio-inspired efficient coding and adaptation," *IEEE Trans. Biomed. Circuits Syst.*, Vol. 6, No. 4, pp. 375 –384, Aug. 2012.

[6] P. Elias, "Universal codeword sets and representations of the integers," *IEEE Trans. Inf. Theory*, Vol. 21, No. 2, pp. 194 – 203, Mar. 1975.

[7] S. Kameda and T. Yagi, "An analog VLSI chip emulating sustained and transient response channels of the vertebrate retina," *IEEE Trans. Neural Netw.*, Vol. 14, No. 5, pp. 1405 – 1412, Sep. 2003.