Improved Lossless Image Compression Using Adaptive Image Rotation

1st Phillip Möller
Deutsche Telekom IT GmbH
Technology Solutions (TSO)
Berlin, Germany
Phillip.Moeller@t-systems.com

2nd Tilo Strutz
Institute of Communications Engineering
Leipzig University of Telecommunications
Leipzig, Germany
tilo.strutz@hft-leipzig.de

***Abstract*—State-of-the-art compression schemes generally have the ability to adapt themselves to the properties of the data to be compressed. This is a kind of learning process and requires the modification of internal variables influencing the treatment of subsequent data segments. In other words: the order of processing has an impact on the compression performance. In image compression, the order can be changed, for example, by rotating the input image by 90°, 180°, or 270°. In application to lossless screen content compression, investigations with different compression schemes (LOCO-I, HEVC, FP8v3, and SCF) have shown that the rotation has a considerable impact on the compression performance. The difficulty, however, is to predict the best rotation. For the SCF (soft context formation) compression scheme, we have developed a method based on a tiny neural network that suggests a suitable rotation by evaluating basic colour properties of the image to be compressed. The compression can be improved by 0.7098% to 1.3817% depending on the image set tested.**

***Index Terms*—lossless image compression, predictive modelling, SCF, processing order, predictive analytics**

I. Introduction

Signal processing schemes, and in particular image compression schemes, usually combine several functional modules in order to produce their output. Each module performs a distinct task it was designed for and passes its output to the next module. The connection of all modules make up the overall compression scheme. The main goal of image compression is to minimize the size of the generated output. A widely used approach to optimize compression performance is the transformation of the input signal. Transformation of the input signal affects the image compression process in two ways: it changes the input for each processing step and it changes the way the course of events takes place because selector modules decide the next execution steps based on a different, transformed signal statistic. For lossless image compression schemes, the transformation of the input signal has to be fully reversible.

Soft-Context-Formation (SCF) [1] is a scheme that has been optimized to effectively compress a class of images called screen content whose importance steadily rises due to the highly increased need to remotely connect clients to cloud services of various kind [2]. One of SCF's main feature is to merge several similar contexts observed up to the current position in the image to create a contextual symbol probability distribution. The contextual symbol probability distribution is required by an arithmetic coder for encoding the current symbol. Each context has a histogram assigned which contains the colours and their frequencies observed in the particular context. When contexts are merged, their assigned histograms are also merged to a single histogram resulting in an estimation about the contextual symbol probability. Each context is formed by the colours of six adjacent pixels A to F around the current pixel X (Fig. 1).

Merging of contexts is done in stage 1 which is the main stage of SCF. If the colour to be coded is not found in the merged histogram, then SCF sends a special (ESC-)symbol and jumps to stage 2 where the symbol probability distribution is estimated by the colour palette of all colours being observed so far. In case that the current colour cannot be found in stage 2, a stage 3 is entered with prior sending of another ESC-symbol. In stage 3, the colour is coded by using a median adaptive predictor (MAP) which separately predicts each colour component. Stage 1 is by far the most effective coding stage as soft contexts can be used to model symbol probability. SCF has been developed and optimized for the coding of screen content which usually contains repeating patterns and has a much lower number of colours than photographs. A high number of colours typically leads to more frequent use of stage 2 and stage 3, making compression less effective.

Researching on SCF, it has been found that the rotation of images before encoding changes compression performance. When the right image orientation is chosen before encoding, savings in compression size are possible. The rotation of an

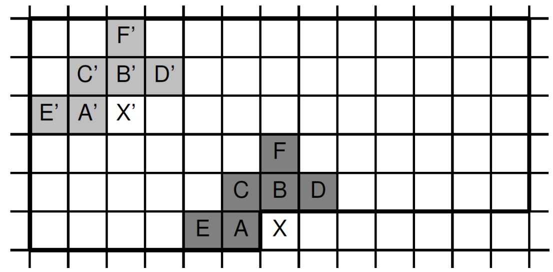

Fig. 1. Pattern around the pixel X and a similar pattern observed at a past position X', according to [1]

978-9-0827-9703-9/19/$31.00 ©2019 IEEE

image before compression can be interpreted as reversible transformation of the input signal. The reason why compression results differ, lies in the fact that the contextual symbol distributions are created differently when the image has been rotated before encoding. Based on image rotation, we have observed improvements in compression performance not only for the SCF coding scheme but also for Fp8v3 [3], HEVC [4] and LOCO-I [5] which is the basic algorithm used by JPEG-LS. The actual challenge is to know which rotation angle leads to the smallest compressed file possible. The naive approach would be to rotate each image in all orientations and to compress it. After compression of all four orientations, the best compression result would be taken. However, this approach is problematic in so far that it increases compression time by factor 4. For compression schemes like LOCO-I, the naive approach is still feasible whereas for more time consuming schemes like Fp8v3, HEVC and SCF overall time consumption would be too high for practical purposes.

This paper shows how the problem of choosing the right image orientation can be solved for the SCF compression scheme. For this purpose, the methods of machine learning and predictive analytics were used. Furthermore, the potential savings in compression costs are provided for Fp8v3, HEVC and LOCO-I to show exemplary that also other compression methods than SCF may profit from image rotation before encoding. It has been found that the optimal image orientations needed for SCF cannot be transferred to other compression schemes as they have different optimal orientations. Nonetheless, the approach presented may be used as a blueprint for other compression schemes to implement a similar functionality.

II. Methods

A. Method for SCF

A set of pictures has been divided into a training set consisting of 97 images and a test consisting of 40 images. The training set (T) was logically divided into T1 (67 images), T2 (10 images) and T3 (20 images) to ensure better comparability to the image sets used in [1] where T1 and T2 were examined. All test images and programs are accessible via [6]. In a first step, for all images of the training set, four versions of each image with different orientations (0°, 90°, 180°, 270°) have been prepared. Those four versions of each image have been compressed and their compressed size has been recorded. The goal was to find the optimal orientation for each individual image and to quantify the maximum savings possible. The savings S were calculated according to:

$$S = \left(1 - \frac{M}{R}\right) \cdot 100 \tag{1}$$

M is the measured compression size of an image after it had been rotated and R is the reference value of the original image. The reference value is the compressed size of an image in case it had not been rotated. The maximum savings possible shall be denoted as S_{max}; it represents the savings in case the encoder would always decide for the optimal orientation of each image. After determination of the optimal orientations of each image, the images of the training set where examined to find out those image properties which can be used by the encoder to bring a decision about optimal orientation. Each image has been divided into four regions of same area which was chosen to be 20% of the whole image size (Fig. 2).

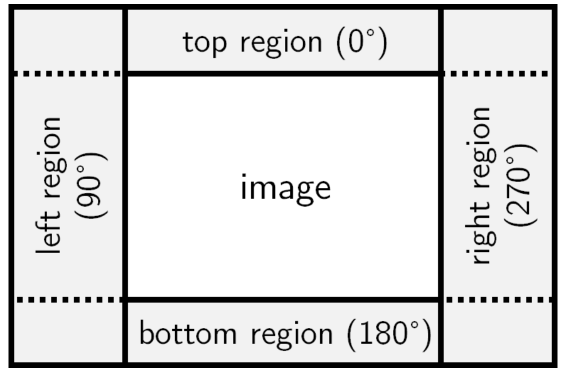

Fig. 2. Image divided into four regions of same area

In the following, those regions will be referred as left, right, top and bottom region. The coding process always starts from the top-left of an image heading to the right. When an image is rotated, the encoder starts in a different region than the top region. So, each region represents the start of the encoder after the image has been rotated by a certain amount of degrees. For example, the left region of an image represents the 90°-rotation as the encoder would start the processing in this region after the image was rotated by 90°.

For SCF, an indicator Q has been defined which is computed in each of the four regions and also for the whole image itself. This indicator expresses the relation between the number of different patterns n_{pat} in a region and the number of colours n_{col} in this region:

$$Q = \frac{n_{\mathrm{pat}}}{n_{\mathrm{col}}} \,. \tag{2}$$

Since the compression performance of SCF is correlated to the frequency of repeating pattern, Q offers an adequate measure of this performance. The lower the number of different patterns is, with respect to a certain number of colours, the more likely the region can be successfully compressed with SCF. So, for each image a quotient vector $\mathbf{Q} = (Q_{\mathrm{l}}, Q_{\mathrm{r}}, Q_{\mathrm{t}}, Q_{\mathrm{b}}, Q_{\mathrm{e}})$ is computed consisting of four elements that are related to the four regions and a fifth element Q_{e} representing the value of the entire image.

Using $\mathbf{Q}$, the prediction of a suitable image orientation is possible. This prediction can be expressed as function $f : \mathbf{Q} \mapsto o$ where o is the suitable orientation of the image.

The prediction itself has been realised based on a feedforward neural network with five input nodes for the five elements of $\mathbf{Q}$ and four output neurons, one for each orientation. The network itself has been implemented using the Genann library which offers a lightweight implementation of a feedforward neural network with a sigmoid activation function [7]. The optimal size of the single hidden layer has been determined based on a systematic search. The goal was to create a network which has a high rate of correct predictions not only for the training set but also for the test set. If the size of the hidden layer is too small, the rate of correct predictions is generally too low. If the hidden layer is too large, the network overfits and has a high rate of correct predictions only for the training set. To prove the generalisation capability of the network, the rate of correct predictions was measured in both the training and the test set. If the rate is nearly the same, it can be concluded that the network has a sufficient generalisation capability.

B. Method for Fp8v3, LOCO-I and HEVC

It has been found that the decisions derived by the neural network optimized for SCF are not transferable to the other compression schemes examined. The reason is that the Quotient Q, by its definition, is a very specific indicator for SCF. For the other compression schemes, merely the optimal rotations and maximum expectable savings have been determined by rotating the images in all four orientations possible.

TABLE I
RATE OF CORRECT PREDICTIONS (%) IN THE TRAINING AND THE TEST SET FOR DIFFERENT SIZES OF THE HIDDEN LAYER

hidden neurons	$\bar{\mathbf{R}}(\mathbf{train})$	$\bar{\mathbf{R}}(\mathbf{test})$	$\mathbf{R}_{\max}(\mathbf{test})$
3	51.65	43.50	57.50
5	64.44	49.75	60.00
7	71.54	53.50	60.00
8	73.52	49.75	52.50
9	74.12	54.25	67.50
10	78.87	48.75	55.00
11	79.49	48.65	55.00
13	84.13	47.50	57.50

TABLE II
COMPRESSION RESULTS FOR SCF BY IMAGE SET

Set	0°	Optimum	Result	$\mathbf{S_{max}(\%)}$	$\mathbf{S_a(\%)}$
T1	2047224	2026170	2032692	1.0284	0.7098
T2	1344347	1320281	1324436	1.7903	1.4811
T3	1363373	1345451	1348895	1.3145	1.0619
Sum	4754944	4691902	4706023	1.3258	1.0288
Test	3223175	3160982	3178641	1.9296	1.3817

III. INVESTIGATIONS

A. SCF

1) Optimal network size: The final neural network has a topology of five input neurons, one hidden layer and four output neurons which represent the four orientations. The size of the hidden layer has been determined empirically. Usually, neural networks are initialized randomly. The consequence is that each newly created network produces different results in regard to rate of correct predictions. In order to determine the optimal size of the hidden layer, it is necessary to calculate the average rate of correct predictions, denoted as $\bar{R}$. Tab. I shows the sizes of the examined hidden layers. For each of those sizes, ten networks have been created and their average performance $\bar{R}$ has been evaluated. For the training set, the the average rate of correct predictions shall be denoted as $\bar{R}(train)$ and for the test set $\bar{R}(test)$.

When the size of the hidden layer increases, $\bar{R}(train)$ increases as well. This is because the network more and more fits to the training data. If $\bar{R}(train)$ increases, $\bar{R}(test)$ decreases as network tends to overfit to the training data. As $\bar{R}$ is an average value, the rate of correct predictions varies around this average. Therefore for each distinct network, the rate of correct predictions might be higher than $\bar{R}$. As generalisation capability should be ensured, $R_{\max}(test)$ is of interest. $R_{\max}(test)$ is the highest rate of correct predictions observed in the test set. The networks having nine hidden neurons performed best as $\bar{R}(test)$ is highest with 54.25% and also $R_{\max}(test)$ is highest with 67.50%. So, the final network chosen has nine hidden neurons. Its rate of correction predictions in the training set is 72.2% and 67.5% in the test set. In 14.4% of the cases the network chooses the second best orientation in the training set. In the test set, the network chooses the second best orientation in 12.5% of the cases.

2) Results: After determining the optimal network architecture, images of the training and test set have been compressed. The SCF version used was a slightly enhanced version of the one in [1]. Tab. II shows the aggregated results of the training set and for the test set. For the whole training set S_{max} is 1.3258%. The actual result, denoted as S_a, using the neural network for decision about orientation is 1.0288%. For the test set S_{max} is 1.9296% and S_a is 1.3817%.

The decisions of the neural network have been taken to conduct a detailed analysis of the progression of coding costs for each stage. It has been found that image rotation effects coding costs in each stage and also how often each stage is visited. Rotation not always has a positive effect on all stages at the same time. If the sum of effects is positive across all stages, savings are also possible. However, it has been difficult to find a clear cause for savings, as many factors affect the compression result. Rotation itself is a simple transformation of the input signal but the effects are complex as the course of events during compression is changed. This means that stages of SCF are entered in a different order which changes the way how the histograms in stage 1 and stage 2 are being created.

The complexity becomes clear when considering how colours and their arrangement affect course of events during compression. A colour is usually part of a pattern and order of events depends on whether the colour has already been observed in the past or whether the colour is part of an already known pattern. In which stage a colour is coded always depends on past signal values (Has the colour been seen before? Is it part of an already known pattern?) and thus also changing future processing order. Those non-linear dependencies are the main reason why a neural network has been chosen to decide about optimal image rotation.

TABLE III
COMPRESSION RESULTS FOR LOCO-I BY IMAGE SET

Set	0°	Optimum	$\mathbf{S_{max}(\%)}$
T1	7630311	7545663	1.1094
T2	6040247	6010531	0.4920
T3	3351967	3345297	0.1990
Sum	17022525	16901491	0.7110

B. LOCO-I

The compression results for LOCO-I can be seen in Tab. III. The maximum saving over the whole training set is 0.7110%. The highest savings by percentage where reached for T1 with 1.1094% where the average number of colours per image is 1645 which is rather low compared to the overall average number of colours in the whole test set (5711). Therefore, it is assumed that the number of colors has an influence on S_{max}.

C. Fp8v3

The compression results for Fp8v3 are given in Tab. IV. The maximum savings possible are 0.8537%. As observed with LOCO-I compression method, the savings for T1 where highest with 1.6163%. So the same hypothesis about influence of number of colours in the image is posed.

TABLE IV
COMPRESSION RESULTS FOR FP8V3 BY IMAGE SET

Set	0°	Optimum	$\mathbf{S_{max}(\%)}$
T1	2132145	2097683	1.6163
T2	1292688	1287788	0.3791
T3	1339643	1338331	0.0979
Sum	4764476	4723802	0.8537

D. HEVC

For HEVC, only T1 and T2 were considered as images in T3 had resolutions not being a multiple of the minimum coding units (CU). The configuration applied for HEVC was the standard configuration of screen-content-coding extension (SCM-6.0, encoder_intra_main_scc_lossless.cfg, InputColourSpaceConvert=RGBtoGBR). Tab. V shows that S_{max} is 0.4132%. The influence of image rotation is lowest within the four compression schemes tested. Nevertheless, rotation has an effect that is probably caused by internal variables of the encoder, which have changing values dependent on the order of processing the pixels.

TABLE V
COMPRESSION RESULTS FOR HEVC BY IMAGE SET

Set	0°	Optimum	$\mathbf{S_{max}(\%)}$
T1	2614838	2602330	0.4783
T2	1496231	1491751	0.2994
Sum	4111069	4094081	0.4132

IV. DISCUSSION AND CONCLUSION

In this paper, it has been shown that image orientation before encoding affects image compression results for a selected group of compression methods which were SCF, LOCO-I, Fp8v3 and HEVC. The rotation of an image is interpreted as reversible transformation of the input signal which changes the order of processing steps for each compression method. For all compression schemes examined in this paper, it has been shown that significant savings in compression size are possible if optimal orientation is chosen by the encoder. The challenge is to extract suitable image properties needed to obtain a decision about optimal image compression before encoding.

For SCF, a method for choosing the optimal image orientations has been presented. This method is based on a trained feedforward neural network which predicts the suitable optimal image orientation on basis of image properties derived by four regions of the image and the whole image itself. The rate of correct decisions made is 72.2% for the training set and 67.5% for the test set. As only 97 images were used in regard to the high time complexity, their number could be increased in future research to increase the rate of correct predictions.

The image compression methods LOCO-I, Fp8v3 and HEVC needed different optimal image orientations before encoding. A transfer of the decisions made by the neural network optimized for SCF is not possible. Also, each compression scheme by itself has different optimal orientations, so that no conclusions about optimal image orientation can be made from one compression scheme to another. For each compression scheme LOCO-I, Fp8v3 and HEVC the maximum possible savings in compression size have been reported. Based on these findings, further research may be conducted on other compression schemes with the goal to implement the same functionality about optimal image orientation before encoding. This also includes compression schemes which were not examined in this paper. The method of dividing the an image in four regions, extracting image properties, and training of a neural network may serve as a blueprint for other compression schemes.

REFERENCES

[1] Strutz T.: "Lossless Intra Compression of Screen Content based on Soft Context Formation" in IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 6, no. 4, pp. 508-516, Dec. 2016.

[2] Pan, Z., Shen, H., Lu, Y., Li, S., Yu, N.: "A low-complexity screen compression scheme for interactive screen sharing" in IEEE Transactions on Circuits and Systems for Video Technology, Vol. 23, NO. 6, June 2013

[3] http://mattmahoney.net/dc/text.html#1532

[4] Sullivan G. J. , Ohm J. R. , Han W. J.: "Overview of the High Efficiency Video Coding (HEVC) Standard" IEEE Transactions on Circuits and Systems for Video Technology , Vol. 22, NO. 12, December 2012

[5] Weinberger, M.J., Seroussi, G., Sapiro, G.: "LOCO-I: A Low Complexity, Context Based, Lossless Image Compression Algorithm" Proceedings of DCC96, 140149

[6] http://www1.hft-leipzig.de/strutz/Papers/SCFrot-resources/ last visited: 12.02.2019

[7] https://github.com/codeplea/genann last visited: 12.02.2019