

# Optimal Quality-Aware Predictor-Based Adaptation of Multimedia Messages

Steven Pigeon and Stéphane Coulombe  
Department of Software and Information Technology Engineering  
École de Technologie Supérieure  
1100 Notre-Dame Ouest, Montréal, Québec, H3C 1K3

**Abstract** – Multimedia Messaging Services (MMS) allow messages composed of different media attachments to be exchanged between heterogeneous devices. In this work, we consider the special case of image-only messages, where the challenge is to adapt messages so that the receiving device constraints are satisfied while maximizing the user’s perceived quality of adapted messages. We propose an adaptation algorithm based on predictors for file size and quality resulting from transcoding parameters that explicitly maximizes an objective function based on the structural similarity image quality index. We show that the proposed method is not only resilient to the imprecision of predictors, but also yields significantly better quality at reduced computational complexity compared to other methods proposed in prior art.

**Keywords** – MMS, image adaptation, JPEG, optimization, predictor, dynamic programming, SSIM.

## I. INTRODUCTION

Multimedia Messaging Services (MMS), allowing users to exchange messages composed of various media attachments, an emerging billion dollars industry [1], require server-side adaptation to ensure interoperability between terminals of different capabilities [2]. Image interoperability issues in MMS are mostly due to image resolutions and file sizes exceeding the receiving terminal’s capabilities, and not to encoding formats as the majority of image traffic is composed of JPEG images.

One must adapt, in the most efficient manner, images both in resolution and file size so that the message satisfies the receiving terminal capabilities while maximizing the perceived quality of the message as a whole. In MMS, the terminal capabilities are characterized by standard profiles [3] such as Image Basic, that limits message size to 30kB and images to maximum resolutions of  $160 \times 120$  pixels, and Image Rich with a message size of 100kB and images up to  $640 \times 480$ , to name a few.

Different solutions have been proposed to adapt images for MMS and other related problems in the wider context of mobile browsing. For example, some solutions minimize transcoding time while satisfying size constraints, but without explicit considerations for quality [4]. Others only apply fixed profiles for transcoding [5]. Still others propose manipulating message structure and images in

order to keep only regions of interest; an approach deemed too complex for realtime MMS adaptation [6], [7].

However, efficiently and accurately estimating the size of a JPEG file resulting from transcoding an image with changes both in resolution and quality factor (a parameter that controls the aggressiveness of compression, and for which we retain the semantics of the IJG [8]) remains a challenge. In previous work, we investigated the issue and proposed predictors and systems for the task of adapting a single image [9]–[11], while in this paper we propose a novel solution using low-cost predictors to optimize multiple image MMS messages explicitly for user experience, subject to the device’s constraints. We show that our solution is capable of finding the optimal solution given a perfect predictor and is quite tolerant of predictors with errors. We also compare our solution to other proposed adaptation algorithms.

## II. PROPOSED SOLUTION

Measuring visual quality of degraded images is a difficult task. PSNR have been used extensively in literature but is ultimately a poor indicator of perceived quality. The structural similarity index (SSIM), proposed by Wang *et al.* is deemed a better indicator of image quality [12]. The SSIM is constrained, without significant loss of generality, to the interval  $[0, 1]$ , with any negative values mapped onto 0. This will allow us to use the following objective function, chosen to balance quality amongst images,

$$Q(M, T) = \prod_{i=1}^n Q(m_i, \mathcal{T}(m_i, t_i)) , \quad (1)$$

where  $M = \{m_1, m_2, \dots, m_n\}$  is the message composed of  $n$  images  $m_i$  with resolutions  $R(m_i) = (w_i, h_i)$ ;  $T = \{t_1, t_2, \dots, t_n\}$  is a series of transcoding parameters  $t_i = (q_i, z_i)$  composed of a target quality factor  $q_i$  and resolution scaling  $0 < z_i \leq 1$ ;  $\mathcal{T}(m_i, t_i)$  the function that transforms an original image  $m_i$  with the transcoding parameters  $t_i$ , yielding an image with a new quality factor  $q_i$  and resolution  $z_i R(m_i) = (z_i w_i, z_i h_i)$ ; and where  $Q(m_i, \mathcal{T}(m_i, t_i))$  is the quality metric (in our case SSIM) between image  $m_i$  and transcoded image  $\mathcal{T}(m_i, t_i)$ . Since the image  $m_i$  and the transcoded image  $\mathcal{T}(m_i, t_i)$  may differ in resolution, the image  $\mathcal{T}(m_i, t_i)$  is rescaled to the resolution of  $m_i$  before comparison [11].

This work was funded by Vantrix Corporation and by the Natural Sciences and Engineering Research Council of Canada under the Collaborative Research and Development Program (NSERC-CRD 326637-05). E-mails {steven.pigeon,stephane.coulombe}@etsmt1.ca

Eq. (1) is to be maximized under the constraints

$$\sum_{i=1}^n S(\mathcal{T}(m_i, t_i)) \leq S(D), \quad (2)$$

where  $S(m_i)$  is the file size of image  $m_i$ , and  $S(D)$  the maximum allowed message size for device  $D$ , and under the orientation-independent resolution constraints

$$\begin{aligned} z_i \max(w_i, h_i) &\leq w_D, \\ z_i \min(w_i, h_i) &\leq h_D, \end{aligned} \quad (3)$$

for each image  $m_i$ , where  $R(D) = (w_D, h_D)$  is the maximum allowed resolution for device  $D$ . Eq. (2) expresses the constraint that the sum of resulting file sizes must not exceed the maximum allowed message size, and eqs. (3) express the constraint that the resolutions of transcoded images must not exceed the terminal's maximum resolution.

This optimization problem can be formulated as a *distribution of effort* problem [13] where resources spent correspond to the file sizes of images, the total budget to the maximum permissible message size, and the objective function is the overall message quality estimated by eq. (1). While it is feasible to maximize eq. (1) exactly by performing a great number of transcodings, it is impractical to do so. Rather than computing eq. (1) and eq. (2) exactly, we will rely on predictors  $\hat{Q}(m_i, t_i)$  and  $\hat{S}(m_i, t_i)$  that estimate the resulting quality and file size, respectively given a characterization of an image  $m_i$  (such as original quality factor, file size, resolution, etc.) and transcoding parameters  $t_i$ , allowing us to rewrite the objective function as

$$\hat{Q}(M, T) = \prod_{i=1}^n \hat{Q}(m_i, t_i) \quad (4)$$

and the constraint in eq. (2) as

$$\sum_{i=1}^n \hat{S}(m_i, t_i) \leq S(D). \quad (5)$$

Eqs. (3) remain unchanged as they contain no uncertainty.

The particular form of eqs. (1) and (4) makes the problem amenable to efficient solutions using polynomial-time dynamic programming [13], which is the proposed approach. The optimal predicted transcoding is given by

$$\hat{T}^* = \arg \max_{T \in \mathcal{T}(M, D)} \hat{Q}(M, T), \quad (6)$$

where the  $T$  are series of transcoding parameters drawn from the set  $\mathcal{T}(M, D)$  of all series of transcoding parameters on message  $M$  that satisfy constraints eq. (3) and eq. (5) of device  $D$ .

The number of transcoding parameters considered for the solving of eq. (6) will depend on the granularity of the predictor as well as on the complexity of the optimization problem one wants to solve, as proportional to the number of images in the message times the number of predicted transcoding parameters; we will further discuss this issue in section V.

### III. TRANSCODING ALGORITHMS

The predictor introduced in [10], [11], hereafter named JQSP (JPEG Quality and Size Predictor), uses the original image quality factor and a file size target to formulate

its prediction on the transcoding parameters and resulting image quality. The predictor was trained on a corpus of 70 000 JPEG images obtained from a Web crawler in 2008 [9]. The density of the predictions is chosen so that, for each original quality factor rounded to the nearest tens, predicted file sizes are at least 5% apart, thus limiting the number of possible transcoding parameters to examine.

To establish the upper-bound of obtainable quality for the proposed predictor-based algorithm, we will use an oracular predictor that returns the exact file size and quality resulting from applying transcoding parameters—the “prediction” is computed by actually performing the transcoding. The oracle will serve to characterize the graceful degradation of the proposed algorithm to the imprecision of the predictors used. Indeed, the oracle can be used to simulate predictors with known error characteristics; and in the experiments we used, in addition to the oracle, oracle-based predictors with 1%, 2%, 5% and 10% relative Gaussian error, 95% of the time. To limit the number of oracular transcoding parameters examined, the oracles yield predictions restricted to quality factors of 10, 20, ..., 100 and scalings of 10%, 20%, ..., 100%.

The proposed optimization algorithm uses predictive dynamic programming to yield a transcoding solution maximizing eq. (4) and satisfying constraints of eqs. (3) and (5). If the predicted solution, once executed, exceeds the size constraint (it will always satisfy the resolution constraints as there are no uncertainties in scaling), the device maximum message size  $S(D)$  is reduced by a factor  $\alpha$  (set to 0.9 in our experiments) and optimization is performed again. The process is repeated until either the constraints are satisfied and transcoding is successful, or the algorithm finds no possible solutions given the predictor and the constraints, in which case the transcoding fails.

Inspired by the fixed profile adaptation in [5], the successive profile optimization algorithm applies successively more restrictive profiles to all images until the transcoded message satisfies the device constraints. For this algorithm, a profile defines the maximum resolution for images as well as the quality factor for compression; the largest profile being, for example, images limited to a resolution of  $640 \times 480$  with a quality factor of 90. The next profile could use the same resolution but a quality factor of 80; each successive profile reducing either resolution, quality factor, or both. The determination of useful profiles is heuristic and depends on the performance objectives one sets. We will see that it may not be useful to have a great number of profiles.

The second comparison algorithm uses a fixed quality factor of 85 and, starting at the maximum resolution allowable for the device, successively scales down images by the same relative factor until the message fits the device constraints. For each image  $m_i$  the largest scaling factor  $0 < z_i \leq 1$  such that  $z_i R(m_i) \leq R(D)$  is found. Adaptation proceeds by adjusting, at iteration  $j = 1, 2, \dots$ , a parameter  $\beta_j$ , starting with  $\beta_1 = 1$ , such that the scaling factor applied to all images becomes  $\beta_j z_i$ . The parameter  $\beta_j$  is adjusted until the resulting images satisfy

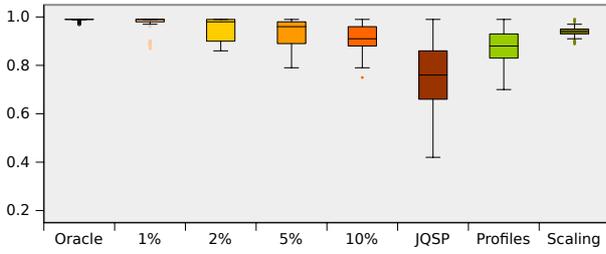


Figure 1. Message capacity distributions, by algorithms and predictors.

the device constraints. For fast convergence, the parameter is adjusted so that  $b_{j+1} = \sqrt{S(D)/S_j}$ , where  $S_j$  is the resulting message size at iteration  $j$ .

#### IV. RESULTS

The JPEG images for the experiments were obtained from the Internet by the use of a Web crawler in the fall of 2010 applied to various high visibility sites. Five JPEG images are randomly picked amongst the 370 000 gathered images to form a message. The experiments show the results for 220 of such messages, of average size of 1.4MB and average image resolution of  $1140 \times 838$ . The target profile for the experiments is Image Rich (images are limited to  $640 \times 480$  and message size to 100kB), thus requiring a 15 : 1 reduction ratio. The dynamic programming algorithm uses the JQSP predictor as described previously, with the predictor trained on the database described in [11] which is disjoint from the 370 000 images used to build the messages.

The resulting capacity, the portion of the maximum message size used by the transcoded message, is shown in Fig. 1. The scores of the objective function for the different algorithms are show in Fig. 2 and the average SSIM for each attachment in a message is shown in Fig. 3.

Fig. 4 shows the objective function scores resulting from individual messages, each curve sorted in ascending order separately. The curves do not allow to compare the relative performance of adaptation algorithms on a same message, but do render the general behavior of the different algorithms and predictors. Fig. 5 presents a similar graph for the average message SSIM.

Fig. 6 shows the times in seconds for the dynamic programming algorithm using the JQSP predictor, the successive scalings and successive profiles algorithms for our single threaded implementation on an Intel T9600 64 bits CPU at 2.8 GHz, using the Magick++ library to perform actual transcodings [14]. Oracular times are not shown, as not applicable to an actual implementation. Table I shows the average number of transcodings and retries (the number of times the algorithm must restart with new constraints) for different algorithms. In our experiment, the fail rate is zero, as all transcodings are eventually successful. For a transcoding to fail, we must have an image large enough so that with a quality factor of 10 and a scaling of 10% it still exceeds the device constraints, and there are no such images in the tested messages.

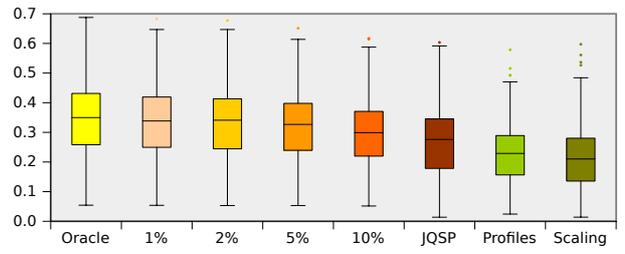


Figure 2. Objective function score distributions, by algorithms and predictors.

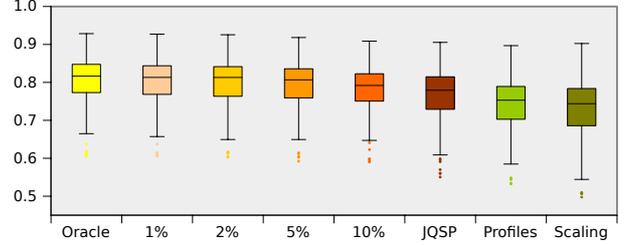


Figure 3. Average SSIM distributions, by algorithms and predictors.

#### V. DISCUSSION

Figs. 1 and 2 show that it does not suffice to maximize capacity to achieve high quality. The great number of profiles used in the successive profiles algorithm allows it to get close to capacity, but the resulting quality is lacking compared to the proposed solution. The successive scalings algorithm gets very close to capacity, but yields worse quality. This is not surprising as neither explicitly maximize message quality, whether expressed as eq. (1) or otherwise. Let us remark that maximizing eq. (1), the products of SSIM scores, for a message is not the same as maximizing the average SSIM score for the same message; but Figs. 4 and 5 show that the two are highly correlated.

Figs. 2 and 5 show that the JQSP predictor behaves close to the 10% relative oracular predictor, and figs. 4 and 5 show that the predictor-based method yields a quality significantly higher than the successive scalings and successive profiles algorithms. These figures illustrate that proposed algorithm is capable of graceful degradation with increasing predictor error.

Fig. 6 shows that the run-time of dynamic programming

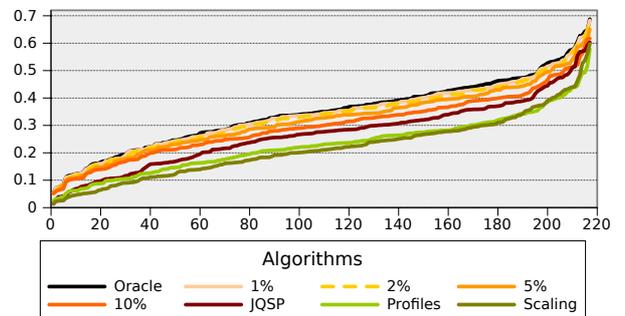


Figure 4. Objective function scores, by algorithms and predictors.

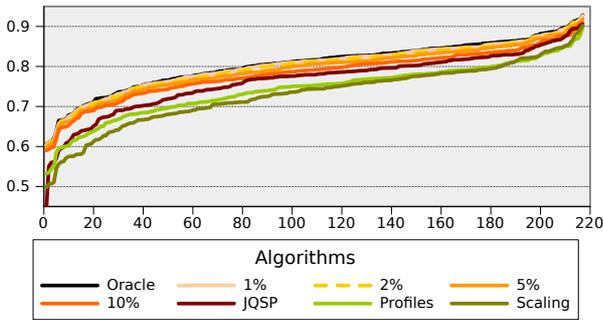


Figure 5. Average SSIM by message, by algorithms and predictors.

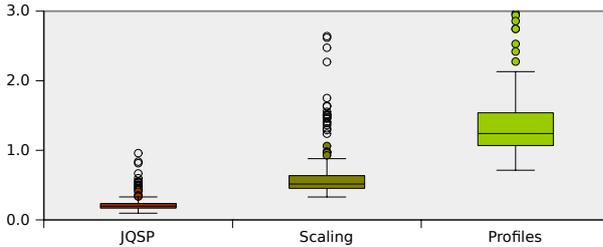


Figure 6. Message adaptation times distributions, by algorithm.

is offset by the gain in fewer transcodings, as to be much faster, on average, with an average of 5.55 transcodings per message, than the successive scalings algorithm with an average of 15.02, and than the successive profiles algorithm with an average of 33.36, as shown in Table I. The slight overshooting of the file size prediction for the JQSP predictor keeps the proposed algorithm from achieving maximal capacity, while still producing higher quality messages, with the side-effect of keeping the number of retries lowest after the oracular predictor; a predictor that undershoots significantly would imply far more transcodings before reaching a solution that satisfies the device constraints.

Furthermore, the number of predicted transcoding parameters examined during optimization plays a non-negligible role in the proposed method performance. Even if the dynamic programming algorithm is efficient, a great number of predicted transcoding parameters per image means a larger graph to explore and necessarily increased run-time, even with pruning, as run-time

Table I. COMPARED AVERAGES OF ALGORITHMS FOR A MESSAGE OF 5 ATTACHMENTS.

Optimization Algorithm	Transcodings	Retries	Objective Function
Oracle	5.00	0.00	0.35
1%	6.03	0.21	0.34
2%	6.54	0.31	0.33
5%	7.19	0.43	0.32
10%	8.25	0.65	0.30
JQSP	5.55	0.13	0.27
Profiles	33.36	5.67	0.22
Scalings	15.02	2.00	0.23

grows quadratically in the number of predicted transcoding parameters [15]. It then becomes a trade-off between the precision of the predicted transcoding and run-time. The successive scalings and successive profiles algorithms are also subject the speed/quality trade-off, but the results are far less interesting. The successive profiles algorithm could use fewer profiles, but already the resulting quality is inferior to the proposed algorithm. The same is true for the successive scalings method, which could start with an aggressive scaling factor of, say,  $\beta_1 = 0.8$ , but that would result only in even worse quality.

## VI. CONCLUSION

The proposed predictor-based dynamic programming message adaptation algorithm maximizes explicitly an objective function based on the SSIM over a complete message rather than on individual images. We show that this strategy does not require the predictors to be almost exact to be useful as we saw that increasing prediction error leads to graceful degradation. In addition, in our experiments, we have shown that our proposed solution is not only much faster, it also yield significantly higher quality for transcoded messages than heuristics found in, and inspired, by prior art.

## REFERENCES

- [1] J. Dwyer, III, "MMS to Prosper as Mobile Marketing Becomes Mainstream," *Wireless Week*, Jan. 2011.
- [2] S. Coulombe and G. Grassel, "Multimedia Adaptation for the Multimedia Messaging Service," *IEEE Communication Magazine*, vol. 42, no. 7, pp. 120–126, July 2004.
- [3] Open Mobile Alliance, "Enabler Test Specification for (Conformance) for MMS Candidate Version 1.3," Oct. 2010, OMA-ETS-MMS\_CON-V1\_3-20101015-C.
- [4] R. Han, P. Bhagwat, R. LaMaire, T. Mummert, V. Perret, and J. Rubas, "Dynamic Adaptation in an Image Transcoding Proxy for Mobile Web Browsing," *IEEE Personal Communications Magazine*, vol. 5, no. 6, pp. 8–17, 1998.
- [5] R. Mohan, J. R. Smith, and C.-S. Li, "Adapting Multimedia Internet Content for Universal Access," *IEEE Trans. Multimedia*, vol. 1, no. 1, pp. 104–114, Mar. 1999.
- [6] W.-Q. Yan and M. S. Kankanhalli, "Multimedia Simplification for Optimized MMS Synthesis," *ACM Trans. Multimedia Computing, Communications, and Applications (TOMCCAP)*, vol. 3, no. 1, Feb. 2007.
- [7] R. C. S. Chen, S. J. H. Yang, and J. Zhang, "Enhancing The Precision of Content Analysis in Content Adaptation using Entropy-Based Fuzzy Reasoning," *Expert Systems with Applications*, vol. 37, pp. 5706–5719, 2010.
- [8] "The independent jpeg group," <http://www.ijg.org/>.
- [9] S. Pigeon and S. Coulombe, "Computationally Efficient Algorithms for Predicting the File Size of JPEG Images Subject to Changes of Quality Factor and Scaling," in *Procs. 24th Queen's University Biennial Symposium on Communications*, 2008.
- [10] S. Coulombe and S. Pigeon, "Quality-Aware Selection of Quality Factor and Scaling Parameters in JPEG Image Transcoding," in *Procs. IEEE 2009 Computational Intelligence for Multimedia, Signal, and Video Processing (CIMSVP)*.
- [11] S. Coulombe and S. Pigeon, "Low-Complexity Transcoding of JPEG Images with Near-Optimal Quality Using a Predictive Quality Factor and Scaling Parameters," *IEEE Trans. Image Processing*, vol. 19, no. 3, pp. 712–721, Mar. 2010.
- [12] Z. Wang, A. C. Bovick, H. R. Sheikh, and E. P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarity," *IEEE Trans. Image Processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [13] F. S. Hillier and G. J. Lieberman, *Introduction to Operation Research*, McGraw-Hill Science, 9th edition, 2009.
- [14] "Imagemagick and magick+," <http://www.imagemagick.org/>.
- [15] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Dover, 1998.