Optimal Quality-Aware Predictor-Based Adaptation of Multimedia Messages

Steven Pigeon
Stéphane Coulombe

Optimal Quality-Aware Predictor-Based Adaptation of Multimedia Messages

Steven Pigeon & Stéphane Coulombe
steven.pigeon@etsmtl.ca
stephane.coulombe@etsmtl.ca

Dept. of Software and IT Engineering
École de Technologie Supérieure
1100 Notre-Dame Ouest, Montréal
Québec, Canada

September 17, 2011

Overview of the presentation

Introduction
- MMS
- Adaptation?

Proposed Solution
- Objectives
- Definitions
- Objective Function & Constraints
- Optimization Problem

Test Methodology
- Messages
- Predictors
- Comparative Algorithms

Results

Conclusion

Problem Definition

What is a MMS?

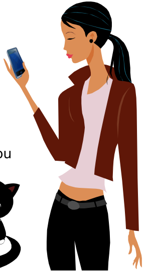
What is a Multimedia Message Service?

- Allows to send multimedia messages
  - Audio, Still Images, Video
- Over Heterogeneous Terminals
- Governed by profiles defining terminal capabilities:
  - Maximum resolution of images,
  - Maximum message size,
  - Image and Video coding standards.

Problem Definition

By Example: *Dramatis Personæ*

Alice



Alice's phone is a
Panaphonics 5SX,
a quite capable phone:

10 MPixel Camera,
Plays media like H.264,
Handles messages
up to 5MB

Minou

On the other hand,
Bob's phone is a
Nokorola Pourave 300,
a quite limited phone:

Does not have a camera
Can display pictures upto 1 MPixel
Handles messages up to 128K

Bob

Problem Definition

By Example: *Dramatis Personæ*

Alice

(Kitty)

Alice's phone is a
Panaphonics 5SX,
a quite capable phone:

10 MPixel Camera,
Plays media like H.264,
Handles messages
up to 5MB

On the other hand,
Bob's phone is a
Nokorola Pourave 300,
a quite limited phone:

Does not have a camera
Can display pictures upto 1 MPixel
Handles messages up to 128K

Bob

Problem Definition

By Example: *Dramatis Personæ*

Alice

Alice's phone is a
Panaphonics 5SX,
a quite capable phone:

10 MPixel Camera,
Plays media like H.264,
Handles messages
up to 5MB

(Koťátko)

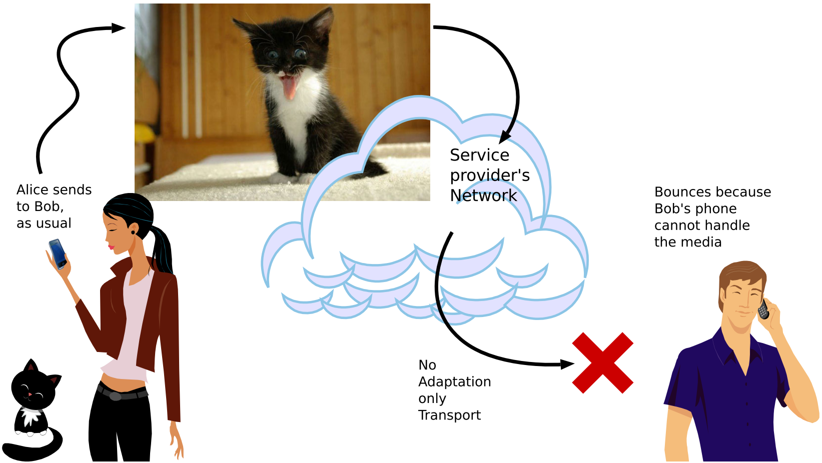
On the other hand,
Bob's phone is a
Nokorola Pourave 300,
a quite limited phone:

Does not have a camera
Can display pictures upto 1 MPixel
Handles messages up to 128K

Bob

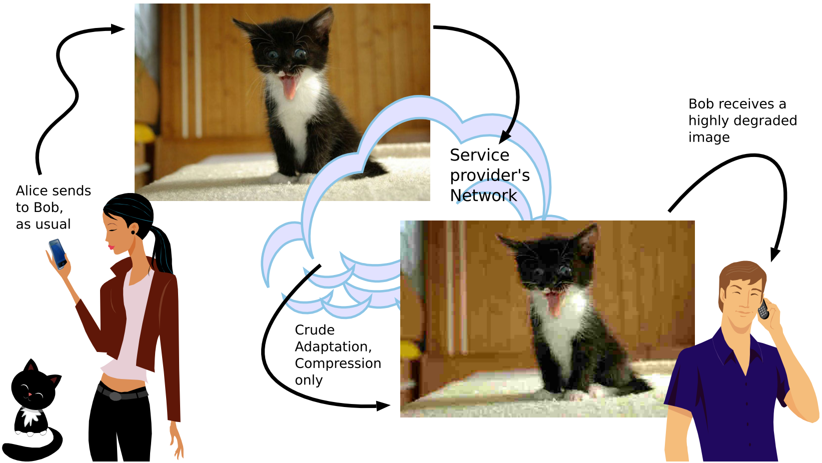
Problem Definition

By Example: Transport Only



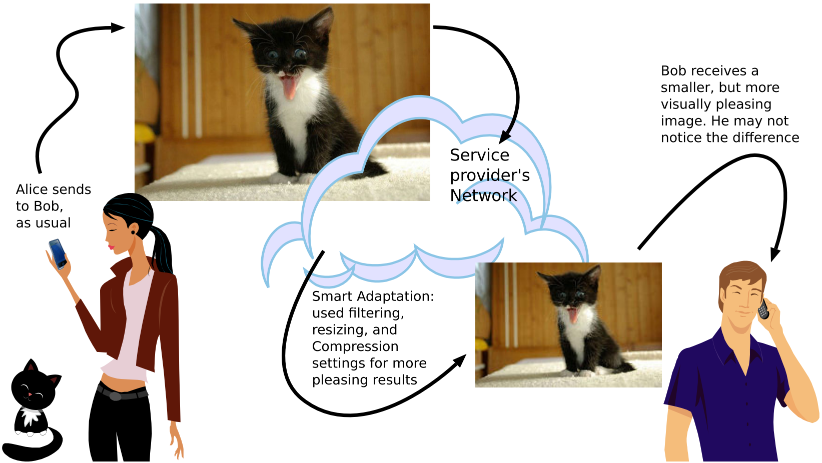
Problem Definition

By Example: Crude Adaptation



Problem Definition

By Example: User-Experience Based Adaptation



Problem Definition

Heterogeneous Terminals

INCOMPATIBLE!

- 960×640 screen resolution
- Essentially no resolution limit
- Essentially no message limit

- 176×220 screen resolution
- "Image Rich" profile
  (640×480 and 100KB messages)

Adaptation

Harder than it Looks

This should be trivial, but...

- The problem is somewhat ill-defined: what is user experience?
  - The problem is amenable to different formulations
  - What is user experience? What are acceptable policies?
- The Problem has to have a computationally efficient solution!
  - While it doesn't look too hard for a single picture, what about multiple pictures in a same message?
  - What if you have to provide this service for a large city? A province? A whole continent?
  - What if you deal with heavier media, like video?

Adaptation

A First Set of Constraints

Smart Adaptation...

- Necessarily server-side (as dictated by MMS)
- Minimizes the computation needed to adapt the media.
- Uses quality metrics to guide optimization, satisfying both constraints (adaptation) and customers (user experience)
- Needs to know a great deal about the media and the target device abstracted as "*viewing conditions*"

Existing Solutions

Fixed Profiles

Some proposed using fixed profiles depending on the receiving terminal*.

...but that's not very good for the users.

Computer

PDA

Phone

* For e.g., Mohan *et al.*

Proposed Solution

Complex Strategies

Smart Adaptation...

- Must be able to change *all* image parameters,
- Must choose best combination of scaling and compression parameters to maximize "user experience" at the message level,
- Must choose a good quality measure, one that correlates highly with the user's perception, say SSIM (which is *much* better than PSNR)

Proposed Solution

Computational Cost

Must minimize computational cost:

- Minimize the number of (tentative) transcodings,
- Must use an *efficient* optimization algorithm,
- and to guide optimization, we will use predictors, algorithms that predict file size and perceived quality of an image subjected to given transcoding parameters.

Definitions

Messages and Images

Let M, be a message composed of n images:

$$M = \{m_1, m_2, \ldots, m_n\}$$

Each image m_i has resolution $R(m_i) = (w_i, h_i)$,

and file size $S(m_i)$.

Definitions

Transcoding Operations

Let T, be the transcoding operations to apply to message M, with $T = \{t_1, t_2, \ldots, t_n\}$.

Each of the t_i describe how to transform image m_i.

We have $t_i = (q_i, z_i)$, where q_i is the new quality factor and z_i the scaling.

Let $\mathcal{T}(m_i, t_i)$ the function that applies the transcoding parameters t_i to image m_i, yielding an image with a new quality factor of q_i and a resolution of $z_i R(m_i) = (z_i w_i, z_i h_i)$.

Definitions

The Receiving Device

Let D, be a receiving device, capable of handling messages of size at most $S(D)$, and images of resolution at most $R(D) = (w_D, h_D)$ (therefore $R(m_j) \leqslant R(D)$, always).

For this work, let us ignore other physical characteristics of the device such as its physical screen resolution and gamut.

Objective Function
SSIM

We will use SSIM, as proposed by Wang *et al.*, as a quality metric.

Let $0 \leqslant Q(m, \widetilde{m}) \leqslant 1$ the function that compares image m and a derivative image $\widetilde{m}$ (a version of m to which was applied some transcoding parameters t).

If $R(\widetilde{m}) \neq R(m)$ (as it would whenever $z \neq 1$), $\widetilde{m}$ is scaled back to the original resolution of m for comparison.

The Proposed Objective Function

The comparison function $Q(m, \widetilde{m})$ will allow us to use the following objective function:

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

which is to be maximized (and has interesting properties!).

Objective Function

...and its Constraints

Eq. (1) is to be maximized under the following constraints:
A size constraint

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

and an *orientation-independent* resolution constraint

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

for each image m_i of message M using the transcoding operations T on device D

Optimization Problem

A Classical Problem

The objective function given by eq. (1) is to be maximized under the constraints given by eqs. (2) and (3), is an instance of a classical optimization problem known as distribution of effort problem.

In this instance, gain is the perceived quality, and the (finite, bounded) resources are the maximum message size, solutions subject to further constraints of resolution.

These problems can be solved quite efficiently with A* search or dynamic programming.

Therefore entirely amenable to efficient algorithms!

Optimization Problem

And Predictors

Maximizing objective function eq. (1) under the constraint of eq. (2) asks to actually perform transcodings for each tentative transcoding parameter series T which is computationally prohibitive, even if we severely limit the values the t_i can take.

The solution is to replace two key components, $Q(\cdot,\cdot)$ and $S(\cdot)$ by low-cost predictors:

That is, replace $Q(m, \mathcal{T}(m,t))$ by the predictor $\widehat{Q}(m,t)$

and $S(\mathcal{T}(m,t))$ by the predictor $\widehat{S}(m,t)$.

Note that now, the expensive $\mathcal{T}(m,t)$ is gone.

Optimization Problem

A Predictor-Based Formulation

The objective function becomes

$$\widehat{\mathcal{Q}}(M,T)=\prod_{i=1}^{n}\widehat{Q}(m_i,t_i) \tag{4}$$

while the size constraint becomes

$$\sum_{i=1}^{n}\widehat{S}(m_i,t_i)\leqslant S(D) \tag{5}$$

and the resolution constraints remain unchanged (there's no uncertainty).

Optimization Problem

Optimizing

We want to solve for the transcoding parameters that (probably) maximize quality, that is:

$$T^* = \arg\max_{T \in \boldsymbol{T}(M,D)} \widehat{\mathcal{Q}}(M, T) \tag{6}$$

where $\boldsymbol{T}(M, D)$ is the set of transcoding parameter series that complies with device D, and

- we will solve using A* or Dynamic Programming,
- we will limit $\boldsymbol{T}(M, D)$ by constraining the possible t_i (say by having $z_i = \{0.1, 0.2, \ldots, 1.0\}$ and $q_i = \{10, 20, \ldots, 100\}$)

Message Generation

Using over 370 000 images obtained by crawling the Internet:

- 220 messages were formed by randomly picking 5 different images,
- with an average size of 1140×838,
- and using "Image Rich" (640×480, 100 KB) as a target.

Predictors

Previous Work

In previous work, we have presented various predictors, but in this work we chose the one presented in *, that we will call JQSP (for JPEG Quality and Size Predictor).

But using only this predictor for tests validates the predictor more than it validates the algorithm!

* S. Coulombe, S. Pigeon — *Low-Complexity Transcoding of JPEG Images with Near-Optimal Quality Using a Predictive Quality Factor and Scaling parameters* — IEEE Trans. Image Processing, V19(3) 2010

Predictors

Oracular Predictors

To validate the algorithm and its behavior, we also used oracular predictors, predictors that always “predict” exactly the resulting file size and quality.

Of course, “prediction” is obtained by performing the actual transcoding, which is expensive.

- With different Gaussian errors: 1%, 2%, 5%, 10% relative error, 95% of the times

Comparative Algorithms

To compare our proposed solution based on dynamic programming, we will use two comparative algorithms:

- "Successive Profiles"
- "Successive Scalings"

both of which are inspired by what is found in literature (and actual products)

Comparative Algorithms

Successive Profiles

The “successive profiles” algorithm will adapt the images by applying successively more restrictive profiles.

For example:

- 640×480, QF=100,
- 640×480, QF=80,
- 320×240, QF=75,
- etc.

...until the message “fits.”

Comparative Algorithms

Successive Scaling

The "successive scaling" algorithm proceeds by successively scaling down the image while maintaining the same reasonable quality factor of 85.

For each image, we find $0 < z_i \leqslant 1$ such that $z_i R(m_i) \leqslant R(D)$, Then a parameter β applied to all images is reduced until the message "fits."

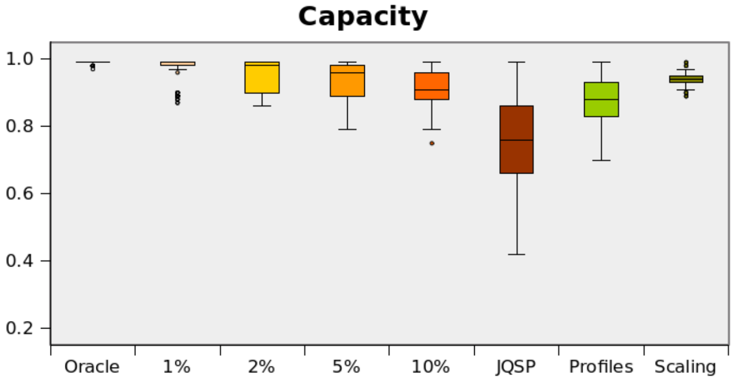
We start with $\beta_1 = 1$, and set $\beta_t = \alpha \sqrt{\frac{S(D)}{S_{t-1}}}$, with $\alpha = 0.95$, where S_{t-1} is the size of the message obtained in the previous step.

Results

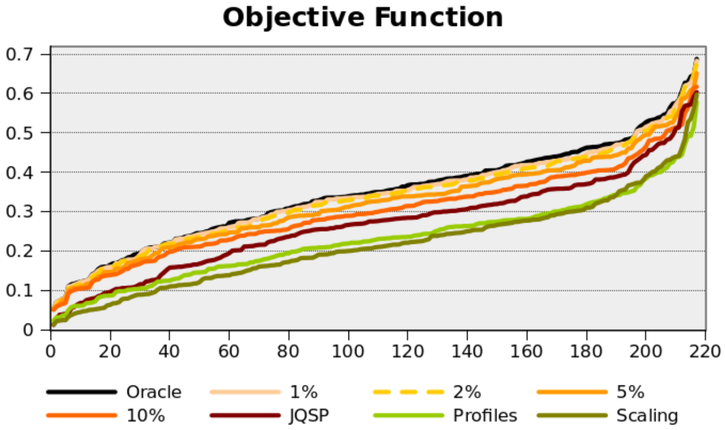
We now present results with 5 images for

- Various predictors:
  - The JQSP predictor,
  - Oracular Predictors (with 1%, 2%, 5% and 10% rel. error)
- All optimizations methods:
  - Dynamic Programming
  - "Successive Profiles"
  - "Successive Scaling"

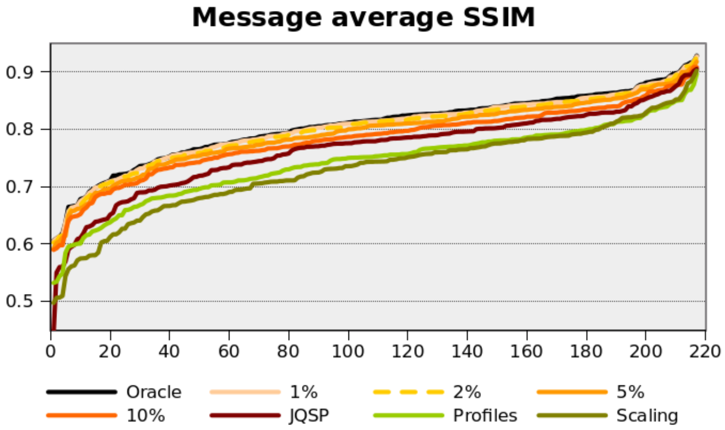
Results



Results


Objective Function
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
20
40
60
80
100
120
140
160
180
200
220
Oracle
1%
2%
5%
10%
JQSP
Profiles
Scaling

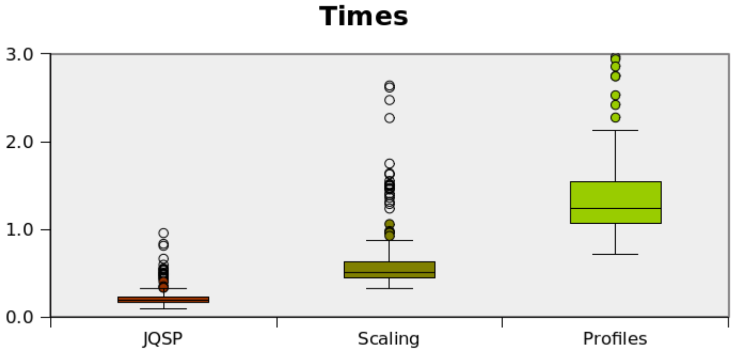
Results


Message average SSIM
0.9
0.8
0.7
0.6
0.5
0
20
40
60
80
100
120
140
160
180
200
220
Oracle
1%
2%
5%
10%
JQSP
Profiles
Scaling

Results

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

Results



(Oracular times are excluded!)

Conclusion

We have seen that the proposed solution...

- Maximizes explicitly user experience,
- Reduces CPU consumption significantly compared to alternative methods,
- Is robust to predictor error.

This work was sponsored by

NSERC CRSNG

http://www.nserc-crsng.gc.ca/

vantrix

http://www.vantrix.com/