Finding a Tradeoff between Compression and Loss in Motion Compensated Video Coding

Thomas Guthier[1,2], Adrian Sosic[1], Volker Willert[1] and Julian Eggert[2]
[1]*Control Theory and Robotics, TU Darmstadt, Landgraf-Georg Strasse, Darmstadt, Germany*
[2]*Honda Research Institute Europe, Offenbach, Germany*

Keywords: Video Coding, Polynomial Motion Model, Quadtree Segmentation, Model Selection.

Abstract: In video coding, affine motion models combined with a quadtree decomposition have often been suggested as an extension to the mostly used translational models combined with a blockwise decomposition. What is missing so far is a thorough analysis to judge the tradeoff between using more complex motion models or more elaborate decomposition methods in terms of data compression and information loss. In this paper, we compare different polynomial motion models with a quadtree decomposition concerning motion model complexity and granularity of decomposition. We provide a statistical evaluation based on optical flow databases to quantitatively find a tradeoff between bitrate and reconstruction error.

1 INTRODUCTION

One of the most important aspects in modern video coding are motion compensation algorithms. Those algorithms segment each image of the sequence and describe the local motion of each segment. This motion information is then used to predict the recent image given the image of the previous timestep. The use of the temporal correlation in image sequences can drastically reduce the bitrate. In lossy video coding, the quality of the compressed video will decrease because of prediction errors of the motion compensation. Therefore the quality requirements must be balanced with bitrate requirements. While video coding standards like MPEG-4 or H.264 (Wiegand et al., 2003) use only block-wise segmentation and purely translational models to describe the local motion, a lot of research focuses on more sophisticated motion models and segmentation methods. An overview on the recent development in video coding can be found in (Sikora, 2005). For example (Zhang et al., 1997) use a hierarchical segmentation including a quadtree decomposition and affine models for motion compensation. Their algorithm shows good results concerning reconstruction quality and bitrate reduction in highly structured scenes, but the bitrate exceeds the coding standards in scences with little motion due to the extra parameters needed for their complex segmentation. (Karczewicz et al., 1997) use a quadtree based segmentation along with polynomial motion models. The quadtree segmentation is easy to implement and needs only one extra bit per segment compared to a regular block-wise segmentation when using an efficient coding as described in (Sullivan and Baker, 1994). Their video coding algorithm showed good results concerning both reconstruction quality and compression, but is not realtime capable due to a complex coefficient selection algorithm which is needed to reduce the number of bits encoding the polynomial motion models. (Lakshman et al., 2010) focus on adaptive motion model selection to overcome the problem of the multiple parameters needed to encode higher order polynomial models.

(a) (b)

Figure 1: On the left is an example image and on the right the corresponding color coded flow field. The color value codes the moving direction and the intensity the amplitude of the motion. The lines show the quadtree segmentation.

Although research is focusing on extensions of the simple translational model and the block-wise decomposition, little has been done to study the complex interdependencies between the reconstruction quality,

Guthier T., Sosic A., Willert V. and Eggert J..
Finding a Tradeoff between Compression and Loss in Motion Compensated Video Coding.
DOI: 10.5220/0004057000810084
In *Proceedings of the International Conference on Signal Processing and Multimedia Applications and Wireless Information Networks and Systems* (SIGMAP-2012), pages 81-84
ISBN: 978-989-8565-25-9

the required number of bits, the segmentation algorithm and the different model orders. The reconstruction quality depends on two elements: The model order and the segmentation. A higher model order and a finer segmentation both increase the reconstruction quality, but then again increase the number of bits. An open question is, whether an increased model order can increase the reconstruction quality and additionally lead to a coarser segmentation and therefore lower the overall required bitrate at the same time. Or if on the other hand, a finer segmentation with simpler motion models yields better results.

The topic of this paper is twofold. The first achievement is a statistical analysis of the gain in reconstruction quality for increasing the order of polynomial motion models. The second achievement is the analysis whether it is a better strategy to spend more bits on the segmentation and less on the model complexity or vice versa. For the purpose of comparison two segmentation algorithms are used. One simple block-wise segmentation familiar to the one used in the MPEG4 standard and a quadtree decomposition. Because this research focuses on motion compensation and not on an entire video coding algorithm, all experiments are directly done on ground truth optical flow datasets and not on image sequences.

2 ALGORITHM

In the following pixel positions in the images are described by $\vec{x} = (x,y)^\top$ and flow fields are given by the corresponding flow vectors $\vec{v}(\vec{x}) = (v_x(\vec{x}), v_y(\vec{x}))^\top$. As a measurement of the reconstruction quality we define the reconstruction error $E_s = \frac{1}{n_s}\sum_{\vec{x}_s} \|\vec{v}(\vec{x}_s) - \bar{\vec{v}}(\vec{p}_s,\vec{x}_s)\|_2^2$. $\|\cdot\|_2$ is the Euclidean norm, $\bar{\vec{v}}(\vec{x}_s) = (\bar{v}_x(\vec{x}_s), \bar{v}_y(\vec{x}_s))^\top$ is the model of the optical flow and $n_s =$ number of pixels per segment.[1] With $s =$ number of segments, we get the normalized reconstruction error $E = \frac{1}{s}\sum_s E_s$.

2.1 Linear Parametric Models

The model order is N, the coefficients for $v_x(\vec{x})$ are $\vec{a} = (a_0, a_1, \ldots)^\top$ and for $v_y(\vec{x})$ are $\vec{b} = (b_0, b_1, \ldots)^\top$. Polynomial models can be described by

$$\bar{v}_x(\vec{x}) = \sum_{n=0}^{N}\sum_{i=0}^{n} a_{n-i,i} x^{n-i} y^i,$$

$$\bar{v}_y(\vec{x}) = \sum_{n=0}^{N}\sum_{i=0}^{n} b_{n-i,i} x^{n-i} y^i.$$

Each segment has its own parameter vector $\vec{p}_s = (\vec{a},\vec{b})$. The number of parameters per model is

$$q_m = 2\sum_{n=0}^{N}\sum_{i=0}^{n} 1 = (N+1)(N+2). \qquad (1)$$

The model of order $N = 0$ is the translational model and has 2 parameters. The first order model is the affine model with 6 parameters. The model parameters are estimated by minimizing the reconstruction error.

2.2 Segmentation

There are two parameters controlling the segmentation process. The quality parameter ε and the maximum segmentation level l_{max}. l is the segmentation level. The algorithm for each segment is:

1. Calculate the model parameters p_s.
2. Calculate the normalized reconstruction error E_s.
3. If $E_s < \varepsilon \quad \vee \quad l = l_{max} \Rightarrow$ stop. Else, continue with step 4.
4. Divide the segment into four rectangluar segments. Increase the segmentation level counter and continue for each new segment at step 1.

One result of this quadtree segmentation can be seen in Fig. 1. For $\varepsilon = 0$ the algorithm has zero tolerance to model errors and is likely to segment the entire flow field into equally sized rectangluar blocks until l_{max} is reached. This is comparable to the block-wise decomposition proposed in the MPEG4 standard.

2.3 Dependency of Segmentation Level and Model Order

Some dependencies have a theoretic nature and can be directly derived from the formulars. In the following, we show under which conditions a quadtree decomposition leads to less parameters than a block-wise decomposition and how increasing the model order and the maximum segmentation level increases the bitrate. Let q be the number of parameters needed to encode one timestep of a motion compensation algorithm. When using block-wise decomposition (index $_b$), no parameter is needed to encode the segmentation. For the quadtree decomposition (index $_q$) one

[1]We choose the reconstruction error instead of the more popular PSNR to distinguish that we compare the flow model to the ideal flow field and not the gray value pixel values with the ones warped by the flow model.

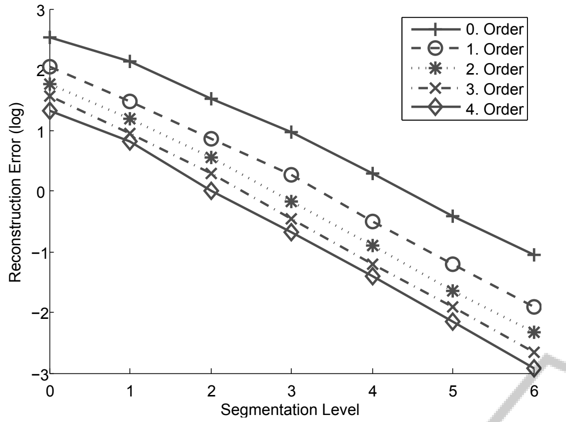

Figure 2: Mean value of the reconstruction error depending on the segmentation level for two optical flow datasets. Each curve shows a different motion model.

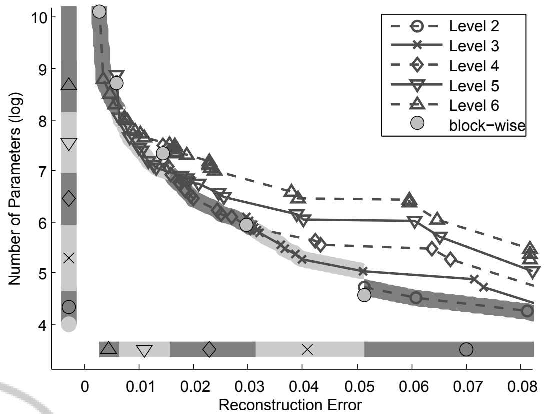

Figure 3: Number of parameters depending on the reconstruction error for different segmentation levels. The algorithm using the affine motion model was applied to the car sequence.

extra parameter per segment is needed for the encoding. We now compare the number of parameters q_q versus q_b with $q_b = s_b \cdot q_m$ and $q_q = s_q \cdot (q_m + 1)$. The inequality $q_q \leq q_b$ leads to $\frac{s_q}{s_b} \leq \frac{q_m}{q_m+1}$. For low model orders the number of quadtree segments s_q has to be smaller than the number of block segments s_b to make the quadtree decomposition more effective than the block-wise segmentation. For higher model orders the fraction $\frac{q_m}{q_m+1}$ converges towards one. Therefore the influence of the extra bit for the quadtree decomposition is decreasing. Next we analyze the increase of q_b due to an increase of N and l for a block-wise decomposition. The number of segments $s_b(l) = 4^l$ exponentially depends on the segmentation level l. With eq. (1) we get $q_b(l,N) = (N+1)(N+2)4^l$ and the gain in q_b:

$$\Delta q_{b,l} = q_b(l+1,N) - q_b(l,N) = 3(N+1)(N+2)4^l,$$
$$\Delta q_{b,N} = q_b(l,N+1) - q_b(l,N) = 2(N+2)4^l,$$
$$\frac{\Delta q_{b,l}}{\Delta q_{b,N}} = \frac{3}{2}(N+1). \tag{2}$$

Incrementing the segmentation level leads to extra bits compared to incrementing the model order, depending linearly on the model order.

3 SIMULATION RESULTS

3.1 Error Depending on Model Order

Segmentation is done with the block-wise decomposition even though the quadtree decomposition shows comparable results. The analysis is done on the entire sequences of the CSAIL (Liu et al., 2008) and Middelbury (Baker et al., 2007) database, which provide ground truth optical flow. The databases contain complex sequences with moving objects as well as homogeneous regions and sequences with little motion. The curves in Fig. 2 show the mean values of all sequences.

The curves are almost parallel throughout the different segmentation levels. Therefore the results are independent of the segmentation quality. Increasing the model order from translational to affine causes the most significant increase in reconstruction quality. From eq. (2) we conclude that incrementing the model order increases the bitrate less than incrementing the segmentation level. We now give an example how this information can be applied to Fig. 2. We start with the translational model (+) at the segmentation level 3. The logarithmic reconstruction error is ≈ 1 and if we want to achieve ≈ 0 we can either increase the segmentation level or the model order by two. Because of eq. (2) latter is preferable.

3.2 Error and Bitrate Depending on the Segmentation Level

In the following we use the affine model. Fig. 3 shows the results for different parameters l_{max} and ε. The algorithm was applied to the highly structured car sequence of the CSAIL database that is shown in Fig. 1. The simulations were performed for the other sequences as well with comparable results. The segmentation algorithm as described in Sec. 2 depends on two parameters, the maximum segmentation level l_{max} and the quality parameter ε. If ε is set a low value little errors are tolerated leading to a stronger segmentation. Larger ε lead to higher reconstruction errors, but less parameters. Each curve represents one l_{max} and different ε, starting with $\varepsilon = 0$. The pareto front is marked. For each bitrate and reconstruction error, the

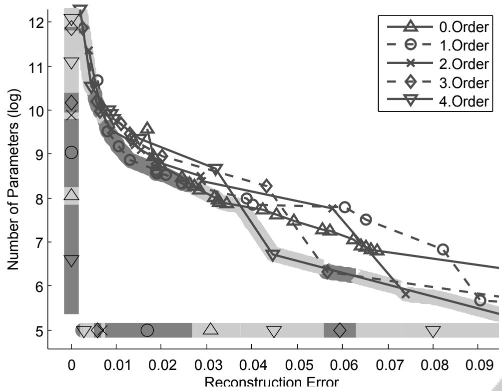

Figure 4: Number of parameters depending on the reconstruction error for different motion models. The algorithm segmented to the fifth level was applied to the car sequence.

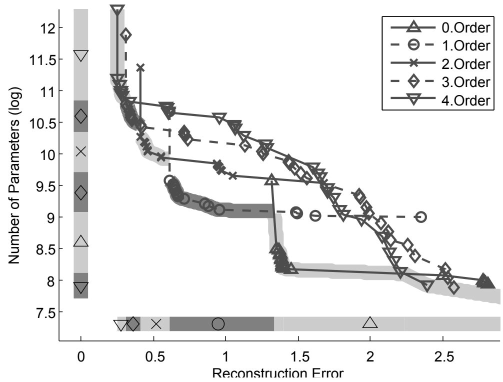

Figure 5: Number of parameters depending on the reconstruction error for different models. The algorithm segmented to the fifth level was applied to the hand sequence.

ideal segmentation level is marked on the pareto front that is additionally plotted next to the axis. The points corresponding to a block-wise segmentation for each l_{max} are additionally marked in Fig. 3.

As discussed in Sec. 2 the block-wise segmentation yields the same reconstruction error as the corresponding graph with $\varepsilon = 0$, but needs less parameters to encode. From the points on the pareto front we can conclude that the desired reconstruction quality or bitrate can be achieved by adapting the segmentation level. There is no overall best segmentation level, rather an ideal level for the different requirements.

3.3 Error and Bitrate Depending on the Model Order

Next we fix the segmentation level $l_{max} = 5$ and compare the different model orders and various quality parameters ε. The results on the car sequence are plotted in Fig. 4. Fig. 5 shows the same algorithm on the hand sequence of the CSAIL database which has on the one hand a lot of motion discontinuities and on the other hand large homogeneous regions. Each curve represents one polynomial motion model and different ε, starting with $\varepsilon = 0$. Like in Fig. 3 the pareto front is marked and additionally plotted with the corresponding model next to the axis. For both sequences there are regions on the pareto front where one model gives the best tradeoff between bitrate and reconstruction quality. Small reconstruction errors refer to larger models.

4 CONCLUSIONS

We provide a qualitative and quantitative analysis for motion compensation in video coding to find the best tradeoff given compression and loss constraints. It is possible to judge which segmentation granularity and motion model complexity best fulfills the coding requirements. The results stress the need for coding algorithms that are adaptive in both the segmentation level and motion model order.

REFERENCES

Baker, S., Scharstein, D., Lewis, J. P., Roth, S., Black, M. J., and Szeliski, R. (2007). A database and evaluation methodology for optical flow. In *Proc. IEEE Conf. ICCV*.

Karczewicz, M., Nieweglowski, J., and Haavisto, P. (1997). Video coding using motion compensation with polynomial motion vector fields. *Signal Processing: Image Communication*, 10(1-3):63 – 91.

Lakshman, H., Schwarz, H., and Wiegand, T. (2010). Video coding with cubic spline interpolation and adaptive motion model selection. In *SPCOM, Conf.*

Liu, C., Freeman, W. T., Adelson, E. H., and Weiss, Y. (2008). Human-assisted motion annotation. In *Proc. IEEE Conf. CVPR*, pages 1–8.

Sikora, T. (2005). Trends and perspectives in image and video coding. *IEEE J.PROC.*, 93(1):6–17.

Sullivan, G. J. and Baker, R. L. (1994). Efficient quadtree coding of images and video. *IEEE J.PROC.*, 3.

Wiegand, T., Sullivan, G., Bjontegaard, G., and Luthra, A. (2003). Overview of the h.264/avc video coding standard. *IEEE Transactions on CSVT*, 13(7):560 –576.

Zhang, K., Bober, M., and Kittler, J. (1997). Image sequence coding using multiple-level segmentation and affine motion estimation. *Selected Areas in Communications, IEEE Journal on*, 15(9):1704 –1713.