Journal of Universal Computer Science, vol. 13, no. 3 (2007), 363-376
submitted: 30/11/06, accepted: 1/2/07, appeared: 28/3/06 © J.UCS

Real-time Architecture for Robust Motion Estimation under Varying Illumination Conditions

Javier Díaz, Eduardo Ros, Rafael Rodriguez-Gomez and Begoña del Pino
(Dep. Arquitectura y Tecnología de Computadores, University of Granada, Spain
{jdiaz;eros;Rodríguez;bpino}@atc.ugr.es)

Abstract: Motion estimation from image sequences is a complex problem which requires high computing resources and is highly affected by changes in the illumination conditions in most of the existing approaches. In this contribution we present a high performance system that deals with this limitation. Robustness to varying illumination conditions is achieved by a novel technique that combines a gradient-based optical flow method with a non-parametric image transformation based on the Rank transform. The paper describes this method and quantitatively evaluates its robustness to different illumination changing patterns. This technique has been successfully implemented in a real-time system using reconfigurable hardware. Our contribution presents the computing architecture, including the resources consumption and the obtained performance. The final system is a real-time device capable to computing motion sequences in real-time even in conditions with significant illumination changes. The robustness of the proposed system facilitates its use in multiple potential application fields.

Keywords: reconfigurable devices (FPGAs), optical flow, real-time image processing, robust illumination systems
Categories: C.1.3, C.3, I.4.8, I.5.4

1 Introduction

The computation of the apparent pixel-motion that is presented in image sequences, (optical flow estimation), is a well known research field which can be used for robotics applications, compression techniques, tracking systems, estimation of 3-D information from motion, etc. There are different approaches based on image block-matching, gradient constraints, phase conservation, or energy models [Barron et al 1994]. Different studies [McCane et al 2001, Liu et al 1998, Barron et al 1994] indicate that gradient models such as the Lucas and Kanade (L&K) approach [Lucas and Kanade 1992] represent a very valid choice with a good trade-off between computing resources requirements and accuracy.

Motivated by these motion estimation comparative studies, we have implemented the L&K model in a coarse grain pipeline structure using reconfigurable hardware [Díaz et al 2006a, Díaz et al 2006d]. Furthermore, we have also implemented an improved version of L&K model [Brandt 1997] adopting a novel design technique for long pipelined architectures [Díaz et al 2006b, Díaz et al 2006c].

This contribution goes one step further and addresses one of the main model hypothesis and limitations: luminance constancy assumption. It is well known [Mesbah 99] that this is a strong assumption usually violated on real-world scenarios and that significantly degrades the accuracy of gradient based approaches. There are

several contributions which face this problem [Kim et al 2005, Treves et al 1994, Chin-Hung et al 2002] and propose complex models based on energy minimization and regularization techniques to deal with luminance variations. The main drawback of these approaches is that their hardware implementation feasibility is difficult. This is so mainly because they are based on global methods which require iterative computations and also because they are based on complex mathematical operations with a high demand on computing resources. In this contribution we propose a new hardware-friendly alternative based on a non-parametric transformation. We have developed a new hardware architecture capable of robustly computing the optical flow under significant luminance variations. As showed in this paper, this method can be efficiently implemented on specific hardware.

The paper has been structured as follows. In [Section 2] we illustrate the modified gradient model. The [Section 3] presents the hardware architecture that we have implemented using reconfigurable hardware (FPGAs) to achieve high-performance real-time computation. The device resources utilization and qualitative results are also presented in this section. Finally, the main conclusions and the future work are given in [Section 4].

2 Rank transform before optical flow computation

In this section we briefly describe the basic L&K optical flow model. We also propose the use of a Rank transform to make the model more robust to varying illumination conditions. The robustness of this new approach is quantified.

2.1 L&K Optical flow model description

Motivated by comparative studies [Section 1], we have chosen the L&K model as a good candidate for real-time optical flow computation. The L&K algorithm is a gradient-based technique. This means that the estimation of pixel velocities is based on image derivatives, and the assumption of constant luminance over a temporal window is utilized. Luminance is approximated by its Taylor series expansion and truncated in the first order. This becomes an ill-posed equation usually known as first order constraint, which is given in equation (1).

$$\left[\nabla_{xy} I(x,y,t)\cdot(v_x,v_y)+I_t(x,y,t)\right]=0 \qquad (1)$$

In equation (1), V_x and V_y stand for the two components of each pixel velocity; I_t represents the temporal derivative of the sequence and ∇_{xy} stands for the spatial (x-y) gradient operator that computes the image derivatives I_x and I_y. There are multiple approaches to overcome the equation (1) limitations. For instance, [Horn and Schunck 1981] build an iterative solution based on the hypothesis of optical flow smoothness. A different solution is presented by Lucas and Kanade who solved this problem by applying least squares fitting on a local neighborhood Ω of each pixel, typically a spatial window of 5x5 pixels [Lucas and Kanade, 1984].

Using this notation, the final velocity estimation is computed from equation (2), where W stands for the weights used to properly integrate estimations over the neighborhood Ω (which typically represents a five taps Gaussian kernel).

$$
\begin{aligned}
V_x &= A_{xy} \cdot A_{yt} - A_{yy} \cdot A_{xt} \\
V_y &= A_{xy} \cdot A_{xt} - A_{xx} \cdot A_{yt} \\
A_{kj} &= \sum_{\Omega} W^2 I_k I_j
\end{aligned}
\tag{2}
$$

We also have adopted in our system the modifications described in [Brandt 97]. Brandt proposes derivative kernels based on complementary derivation-smoothing operations that significantly improve the derivation process accuracy. Furthermore, this leads to reducing the temporal pre-smoothing filter length from 15 to 3 taps, which is important for embedded systems and also helps to enhance the accuracy of the model.

According to the previous discussion and equation (2), the main steps required for this model can be summarized as follows:

- S_0: Spatio-temporal image smoothing based on FIR Gaussian filters.
- S_1: Temporal smoothing and derivative operations.
- S_2: Spatio-temporal image derivatives computation from temporal data based on first order Gaussian kernels.
- S_3: Image derivatives combination on a local neighborhood to construct the basic elements for least squares fitting (A_{kj} elements on equation (2)).
- S_4: Estimation of local velocity values.

This is the basic model that we will take as reference. Hereon we call it the L&K model although we included the above mentioned modifications proposed in [Brandt 97]. In the next section we evaluate its behavior under varying illuminations conditions.

2.2 Optical flow using Rank transformation

The formulation of equation (1) assumes that an image pixel, representing a point on an object, does not change its brightness value from an instant t to the next instant $t+\delta t$. However, in a realistic scene this is almost never the case. A pixel can change its brightness value because an object moves (translates or rotates) to another part of the scene with different lighting or because the illumination of the scene (globally or locally) changes in time.

An alternative to the regularization methods described in [Kim et al 2005, Treves et al 1994, Chin-Hung et al 2002] is to perform a non-parametric image transformation. This has already been used as an effective technique for stereo matching in [Bank et al 1997]. But to the best of our knowledge, this method has never been used for estimation of optical flow with gradient-based approaches. Non-parametric techniques, for instance Rank and Census transforms, are described in [Zabih and Woodfill 1994]. They are based on the relative ordering of pixel intensities within a window rather than the intensity values themselves. Consequently, these techniques are robust with respect to (global or local) illuminance variations. Furthermore, differences in gain and offsets between frames of the sequence will not affect the ordering of pixels within a window. This represents an effective technique to prevent multiplicative and additive illuminance noise problems. They are also tolerant to a small number of outlying gray levels within the window and, therefore,

they are also robust with respect to small amounts of random noise [Bhat and Nayar 1996].

In our contribution we use the Rank transform. This is defined as the number of pixels in the window whose value is smaller than the central pixel. For example, a window of 3 × 3 pixels is presented in expression (3). The symbol ζ stands for the Rank transform itself. The value is 5 because there are 5 elements in the window whose values are smaller than the one in the central position.

$$\zeta\begin{pmatrix} 27 & 26 & 18 \\ 19 & 23 & 20 \\ 17 & 21 & 25 \end{pmatrix} = 5 \tag{3}$$

In this contribution we have used a window of 11x11 pixels. This technique is hardware friendly because it can be implemented using a small memory buffer of 11 image rows and a comparator tree. In our system, we transform the input image stream using this technique and after that we apply the L&K model. Hereon, we note RL&K to the method that computes optical flow using the L&K technique over the Rank transformed images. It is important to note that L&K method is feasible on Rank transformed images because continuity and derivability properties are preserved by this transformation.

We shall also take into account the optical flow confidence measure. L&K requires a confidence estimator to reject image areas with unreliable information [Bainbridge-Smith and Lane 1996]. Our estimator of confidence, noted as C_e, is described in equation (4).

$$C_e = A_{xx} \cdot A_{yy} - A_{xy}^2 \tag{4}$$

Image patches with low contrast produce low C_e values and can be easily rejected using a global threshold. One limitation arises when we apply the rank transformation because luminance information is lost. Therefore, to properly reject image areas of low contrast we adopted the following strategy. We compute the C_e from the input image as well as its rank transformation. After that, we threshold the rank values using the C_e information to reject low confidence image areas. If we implement this in software this means that the model becomes slower and therefore less suitable for real-time systems. Nevertheless, since we are using FPGAs technology we can replicate some computing units, increasing the resources consumption but without any penalty in terms of performance. This technique has been successfully applied in our system as described on [Section 3].

2.3 Quantitative evaluation of the RL&K model robustness to varying illumination conditions

We have numerically evaluated the accuracy of both approaches under varying illumination conditions using the synthetic sequence of the through-flow across the Yosemite Valley whose real ground-truth is known (Sequence available at [VIS]). We used the error measurement of [Fleet 1992] and a C_e value that produces an estimation density of 50% of the image pixels.

We have synthetically generated several varying illumination conditions that we note T*emporal Illuminance Patterns* and that are illustrated in [Fig. 1]. We took 13 frames of the Yosemite Valley sequence and modified their global luminance values

according to equation (5). Note that optical flow is computed only for the central image (number 7) and that the remaining frames are used in the temporal smoothing and derivative operations. In equation (5), *I'* stands for the new pixels luminance values and *I* for the original ones. IV_{GAIN} represents the *Illuminance Variation Ratio* which in our experiments goes from 0 to 0.4. This represents up to a 40% of illuminance variation between central frame and the remaining ones. T_P stands for the illuminance factor which is used for each frame according to the 5 temporal patterns illustrated in [Fig. 1]. Finally, the $\|.\|_{[0,255]}$ notation stands for the truncation operation, necessary to rescale the data to the original range of 0 to 255 and simulate the luminance saturation camera effects.

$$I'(k) = \left\| I(k) \cdot \left(1 + T_p(k) \cdot IV_{GAIN}\right) \right\|_{[0,255]} \quad (5)$$

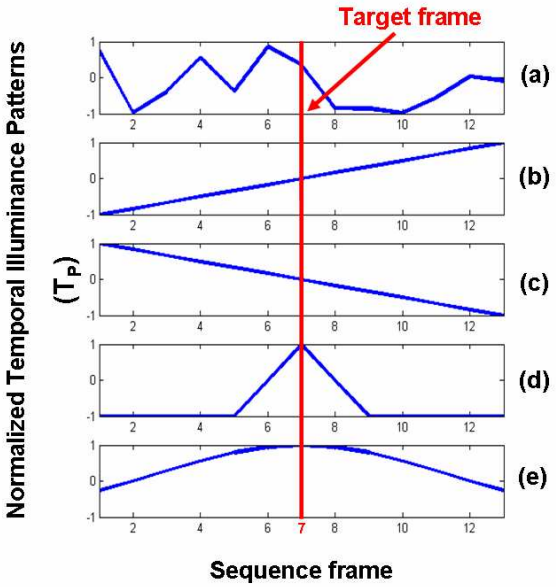

Figure 1: Normalized Temporal Illuminance Patterns used in our experiments. (a) Completely random illuminance variation. (b) Increasing illuminance ramp. (c) Decreasing illuminance ramp. (d) Delta illuminance variation. (e) Smooth variation of the illuminance conditions. Note that optical flow is computed only for the central frame (number 7) and that the reaming ones are used for temporal smoothing and derivation operations. The new luminance sequence values are computed using this pattern and the equation (5).

[Fig. 1] presents 5 different temporal patterns which have been used for the evaluation of the optical flow robustness. The first pattern (a) is a completely random illuminance variation, (b) and (c) represent increasing or decreasing illuminance ramps. Finally, patterns (d) and (e) respectively represent a delta and smooth variation of the illuminance conditions.

These patterns have been applied to the Yosemite sequence and used for the optical flow computation using L&K and RL&K models. The robustness has been evaluated using the angular error of the computed flow with the known real ground-truth. As expected, the results presented in [Fig. (2)] confirm that RL&K model is significantly robust to illuminance variations while L&K model is highly affected by these artefacts. Optical flow using *L&K* model has an angular error of 5.1º for a density of 50% but, for some patterns, the error can increase in more than 550%. Even for small IV_{GAIN} values of 0.1, the angular error can increase in more than 100%. On the other hand, *RL&K* model has an initial error of 7.3º. But this approach shows a very stable behaviour for IV_{GAIN} values up to 0.25 and in the worst case (IV_{GAIN}=0.4, pattern (d)) the maximum error variation is, 1.8º, (an increase ratio below 25% of the initial error).

Note that the models behave differently in the presence of illuminance temporal variation patterns. *L&K* shows the worst behaviour for the random stimulus, pattern (a), whereas the *RL&K* model leads to the worst results when applied to the pattern (d). In some cases (for instance the RL&K model in the presence of pattern (e)) the accuracy even increases with the illuminance variation. Nevertheless this can be easily understood since the confidence values used for the two methods are not the exactly the same and even the confidence pixels can change across the different IV_{GAIN} values. The image areas with C_e values above a predefined threshold depend on each value of IV_{GAIN} and therefore, optical flow density slightly varies for each configuration (typically between 45% and 55%). Furthermore, rounding operations to keep data in the [0,255] range also modify confidence results. In our error estimation global values, we have linearly scaled them according to density variation to take into account this effect but this is a non-linear process. This is the reason for the two optical flow models showing a different behaviour when facing different illuminance variations artefacts. However, despite these different behaviours the main conclusion of this comparative study is that RL&K model is significantly stable under illuminance varying conditions.

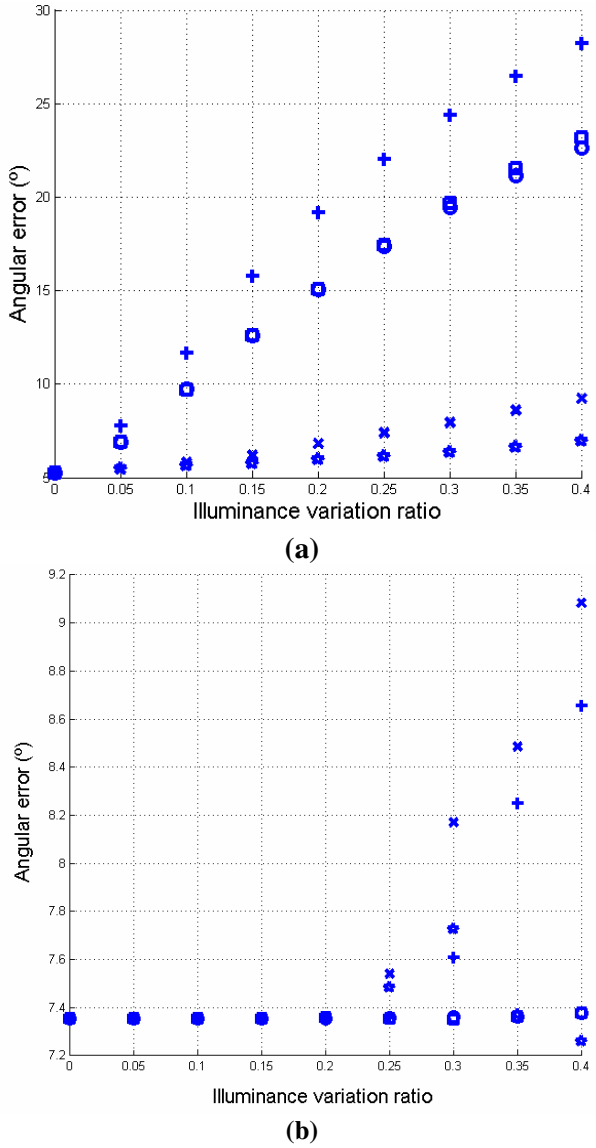

Figure 2: Robustness evaluation under varying illumination conditions of the optical flow gradient based techniques: (a) L&K and (b) RL&K models. The results related to the different temporal patterns are represented using different plot marks: x for the pattern (a), squares for (b), circles for (c), crosses for (d) and stars for (e). Note that, though L&K model is quite accurate in absence of illumination variations, this is significantly degraded with illuminance varying patterns. On the other hand, RL&K produces less accurate results but its behavior under illumination variations is very stable.

3 Robust optical flow computing architecture

We have implemented the RL&K optical flow approach in hardware on a Virtex II XC2V6000-4 FPGA. The system has been designed in a stand-alone processing platform (RC300 board from Celoxica, [CEL]) that can be used on mobile applications. Thus, we have embedded on the same device also the user interface, hardware controller for memory, VGA visualization and input camera frame-grabber. We have used Handel-C [HAN 2003] as hardware specification language. DK software [CEL] as synthesis tool to generate the Edif input to the Xilinx ISE environment [ISE], which has been used as FPGA place and route software.

One of the important improvements of the Rank transform over other literature contributions is that it can be incorporated as a preliminary stage prior to optical flow computation. Adopting a modular design strategy this allows the straightforward use of a previous high performance architecture described in [Díaz et al 2006b, Diaz et al 2006c]. That system computed the optical flow using the L&K model. It has been designed using large pipeline structure with more of 70 stages achieving real-time computation beyond the 150 fps at VGA resolution. The main limitation of this system comes from the optical flow model in which it is based, unable to deal with variations on illuminance conditions. This means that, even simple auto-gain of the cameras leads to noisy motion estimations since the constant illuminance assumption of the model is not fulfilled.

The same architectural strategy that we adopted for the L&K system has been used for our RL&K implementation, see [Section 3.1]. According to the description of [Section 2], the required computation can be summarized as follows:

a) S_{-1}. Pre-processing. Image Rank transformation and computation of local confidence estimator (using equation (4)). Both are computed in parallel.
b) S_0. Gaussian-filter smoothing stage.
c) S_1. FIR temporal filter for computing the temporal derivative and space-time smoothed images.
d) S_2. Spatial derivatives and complementary Gaussian filtering operations.
e) S_3. Construction of least-square matrices for integration of neighborhood velocities estimations.
f) S_4. Custom floating-point unit. Final velocity estimation requires the computation of a matrix inversion, which includes a division operation. At this stage the resolution of the incoming data bits is significant and expensive arithmetic operations are required. Thus fixed-point arithmetic becomes unaffordable, prompting us to design a customized floating-point unit.

The first stage is noted as S_{-1} to indicate that this is a pre-processing stage to make the system robust to illuminance variations and that is computed before the optical flow.

3.1 Basic architecture details

The basic computational stages are presented in [Fig. 3]. This figure schematically presents the different steps of RL&K algorithm described on the introduction of [Section 3].

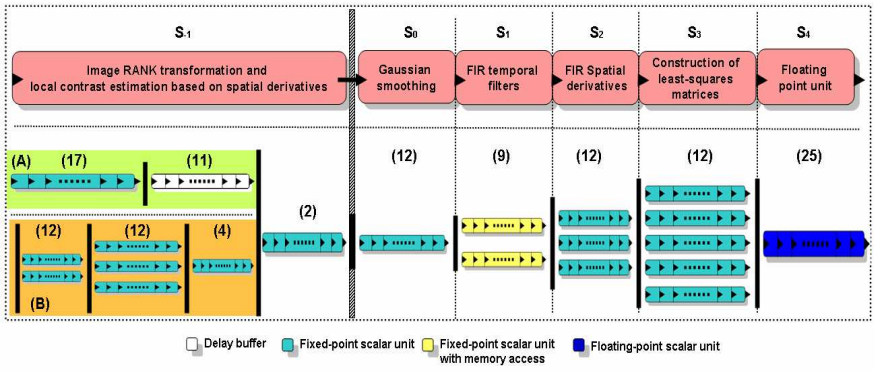

Figure 3: RL&K optical-flow processing core. Coarse pipeline stages are represented at the top and superpipelined scalar units at the bottom. Gray levels indicate the type of scalar units according to the legend at the bottom of the figure. The number of parallel datapaths increases/decreases according to the structure of the optical-flow algorithm. The whole processing core has more than 100 pipelined stages allowing the computation of one optical-flow estimation per clock cycle. The number of substages for each coarse-pipeline stage is indicated in brackets.

The legend of [Fig. 3] indicates the internal data representation of the scalar units. The number of parallel units is driven by the intrinsic parallelism of the model. Note that the level of parallelism is only schematically represented at each stage. There are some internal operations computed in parallel at each scalar unit to get a throughput of one estimation per clock cycle (therefore some of these datapaths can be seen as superscalar units). We used fixed-point representation for all the stages but S_4, which uses floating-point representation. This was a critical decision motivated by the large incoming bit-width at this stage, which makes fixed-point representation very expensive in terms of computational resources. This topic is amply discussed in [Díaz et al 2006b, Diaz et al 2006c]. The floating-point unit has been completely designed using Handel-C language [HAN 2003], with a pipelined architecture. The design is parametrizable, allowing customized mantissa and exponent bit-width values.

In [Fig. 3] we can see that there are two different datapaths at the pre-processing stage S_{-1} noted as (A) and (B). The processing unit (A) performs the Rank transformation of the image. Datapath (B) computes the confidence estimator following equation (4). This unit has been developed using the computing primitives developed for stages S_2 and S_3, allowing code reutilization and fast hardware design. The last unit (two pipelined stages) of S_{-1} is used to reject low confidence image regions.

In stages S_0-S_4, we use the L&K model almost without requiring modifications. We only have included some logic at the computing stages to detect unconfident data values (represented by a null value out of Rank transformation range) and avoiding the propagation of their results. This is necessary because computation uses information available in small neighbourhoods of several pixels and, therefore, non reliable and reliable data can be available in the same window at the same time. This can cause wrong estimations, especially on boundaries between low-contrast and

high-contrast areas that shall be avoided. Based on the central window pixel value, a decision tree processes each pixel or assigns the null value for further computations. This null value indicates that it is a low confidence measure that shall be discarded without stopping the pipeline.

3.2 System resources utilization and final performance

In [Tab. 1] are shown the hardware costs of the whole designed system (processing motion core, MMUs, frame-grabber, VGA output and user-interface) and its performance. First row presents the results for the architecture based on the L&K model and the second row shows the resources for the RL&K based system. Note that, though performance is the same (our clock rate limiting stage is S_4), the resources utilization significantly increases for the RL&K based architecture (from 27% to 40% of the FPGA resources). This is due to the massive parallelism required in our design to maintain a high throughput without any performance degradation (with respect to the L&K system). Nevertheless, depending on the target application, we can share resources reducing the device utilization percentage.

Slices / (%)	EMBS / (%)	Embedded multipliers / (%)	Mpps	Image Resolution	Fps
9319 (27%)	28 (19%)	12 (8%)	50.1	640x480	163
13579 (40%)	53 (36%)	12 (8%)	50.1	640x480	163

Table 1: Complete system resources required on a Virtex II XC2V6000-4 after place and route for the L&K (first row) and RL&K (second row) models. The system includes the optical flow processing unit, memory management unit, camera frame-grabber, VGA signal output generation and user configuration interface. (Mpps: mega-pixels per second at the maximum system processing clock frequency, EMBS: embedded memory blocks).

The image resolution can be selected according to image input camera specifications or processing capabilities. This architecture is scalable being possible to reduce the system parallelism (and performance) to fit on a smaller device. Furthermore, the processing core can be replicated (more than 60 % of system resources of a Virtex II XC2V6000 are available) to split the image and send it to several processing units. This high level scalability allows multiplying the processing performance on this board but since the architecture already fulfils the requirements to compute in real-time sequences at a high frame-rate we have not addressed this issue.

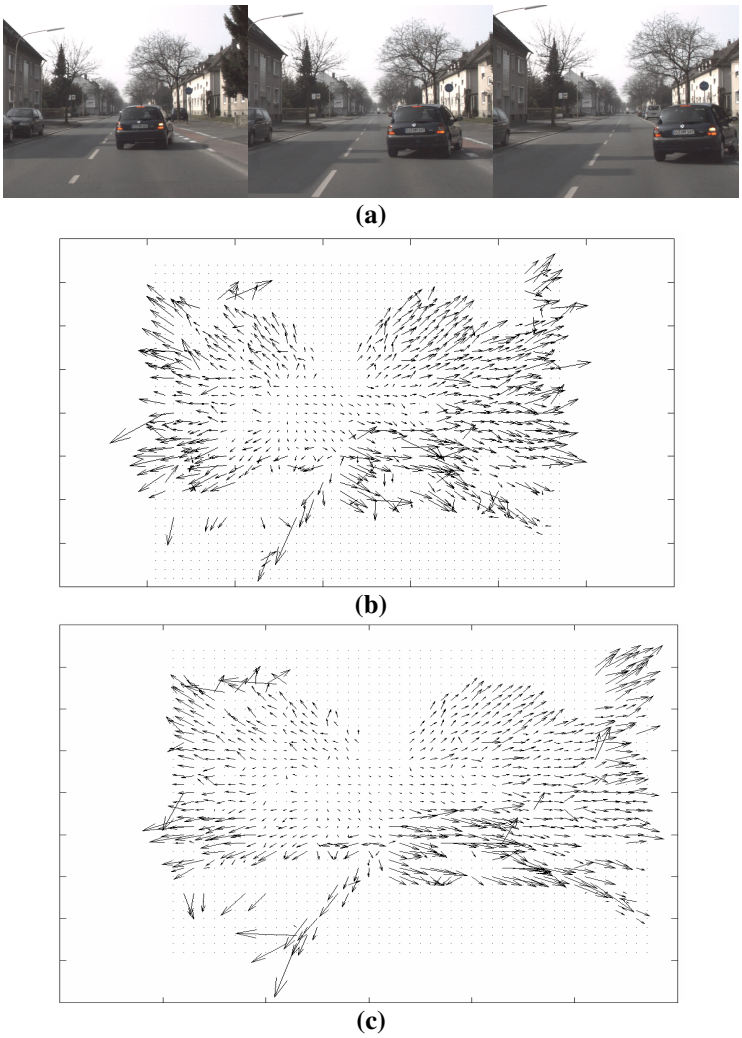

Figure 4: Optical flow results. (a) Three frames of an input image sequence taken from a car moving forward. The car in front of the camera is turning to the right. (b) Optical flow computed with the L&K based system. (c) Optical flow computed with the RL&K system. Velocities are encoded using oriented arrows of size proportional to the speed. From the expansion focus, velocity vectors are approximately radial, with larger velocities in the image periphery. Note that results from L&K and RL&K are very similar though small differences can be found, for instance in the road mark area.

In order to qualitatively evaluate both approaches, [Fig. 4] shows the optical flow results represented using oriented arrows for a real-world driving scenario. In [Fig. 4.(b)], the optical flow is computed with the system using the L&K model. In [Fig.

4.(c)], the RL&K architecture is used. These results show that the optical flow accuracy of both approaches is similar although slight variations are visible.

4 Conclusions

This contribution proposes the use of the Rank transform as a hardware friendly pre-processing that makes gradient-based optical flow models robust to illuminance variations. In order to evaluate this, we have compared the L&K model with the RL&K method in which we apply the Rank transform before the L&K model. We measure the robustness of the two approaches to illuminance variation patterns. The RL&K shows a much more robust behavior to varying illuminance conditions. The work compares quantitatively in terms of accuracy the behavior of the L&K model with the proposed RL&K approach.

We present the hardware implementation of the proposed scheme. The design of a fine grain pipelined computing scheme of the RL&K model allows the same performance as the original model at the cost of hardware resources. We have also implemented the calculation of the optical flow confidence measure replicating some circuitry at the pre-processing stage since the luminance values in which it is based are lost after the Rank transform. This again requires further hardware resources but does not represent performance degradation in the fine grain pipelined processing datapath.

The model has been successfully implemented in hardware using a XC2V600-4 Virtex II FPGA. A massive parallel architecture has been designed for this purpose. The Rank transformation has been chosen as a hardware friendly approach which suits well on reconfigurable devices. A large pipeline of more than 100 stages and multiple scalar units allows us to achieve an outstanding computing performance of 163 fps at VGA resolution. Although the computing resources are quite high (40% of the used device), the adopted scheme can take full advantage of resources sharing techniques to reduce hardware utilization when we require a lower performance.

The presented approach is particularly appropriate in scenarios in which the scene illuminance varies significantly. For instance in automobile applications sequences taken from on-board cameras show rapid illuminance variations due to other cars lights (during night driving) or due to shadows when driving during the daylight.

As future work, we plan to study resource sharing techniques to fit the design into smaller devices. We are also working on a multiscale version of the RL&K model to improve the velocity range of the model. Further experiments to evaluate the robustness of the Rank transformation and its applicability to other optical flow approaches are also part of our future work.

Acknowledgements

This work has been supported by the grants DEPROVI (DPI2004-07032), DRIVSCO (IST-016276-2) and TIC2007:"Plataforma Sw-Hw para sistemas de visión 3D en tiempo real".

References

[Bainbridge-Smith and Lane 1996] Bainbridge-Smith, A. and Lane, R.G.: "Measuring Confidence in Optical Flow Estimation"; IEE Electronic Letters, 10, (1996), 882-884.

[Banks et al 1997] Banks, J., Bennamoun, M., Corke, P.: "Non-parametric techniques for fast and robust stereo matching"; TENCON '97. Proc. of IEEE Region 10 Annual Conf. Speech and Image Technologies for Computing and Telecommunications', 1, (1997), 365-368.

[Barron et al 1994] Barron, J.L., Fleet, D.J., Beauchemin, S.: "Performance of optical-flow techniques"; International Journal of Computer Vision, 12, 1 (1994), 43-77.

[Brandt 1997] Brandt, J.W.: "Improved Accuracy in Gradient Based Optical Flow Estimation"; IJCV, 25, 1 (1997), 5-22.

[CEL] RC300/RC340 Desktop Platform and DK software suite. Celoxica Company, http://www.celoxica.com.

[Chin-Hung et al 2002] Chin-Hung, T., Shang-Hong, L., Yung-Sheng, C., Wen-Hsing, H., "Robust computation of optical flow under non-uniform illumination variations"; Proc. of 16th International Conference on Pattern Recognition, 1, (2002), 327- 330.

[Díaz et al 2006a] Díaz, J., Ros, E., Pelayo, F., Ortigosa, E. M., Mota, S.: "FPGA based real-time optical-flow system"; IEEE Transactions on Circuits and Systems for Video Technology, 16, 2 (2006), 274-279.

[Díaz et al 2006b] Díaz, J., Ros, E., Mota, S., Rodriguez-Gomez, R.: "Highly paralellized architecture for image motion estimation"; Lecture Notes in Computer Science, Springer-Verlag, 3985, (2006), 75-86.

[Díaz et al 2006c] Díaz, J., Ros, E., Rodríguez-Gómez, R., del Pino, B.: "Análisis y diseño de una arquitectura súpersegmentada de altas prestaciones para estimación de movimiento"; VI Jornadas sobre Computación Reconfigurable y Aplicaciones (JCRA 2006), Cáceres, (2006), 275-280.

[Diaz et al 2006d] Díaz, J., Ros, E., Mota, S., Pelayo, F., Ortigosa, E. M.: "Sub-pixel motion computing architecture"; IEE Proc. Vision, Image & Signal Processing, 153, 6 (2006), 869-880.

[Fleet 1992] Fleet, D.J.: "Measurement of Image Velocity"; Engineering and Computer Science. Kluwer Academic Publishers, Norwell, MA, (1992).

[HAN 2003] Handel-C language reference manual, version 3.1, document number RM-1003-3.0. Celoxica Company, 2003.

[Horn and Schunck 1981] Horn, B.K.P, Schunck, B.G.: "Determining optical flow"; Artificial Intelligence, 17, (1081), 185-203.

[Kim et al 2005] Kim, Y.H., Martinez, A. M., Kak, A. C.: "Robust Motion Estimation under Varying Illumination"; Image and Vision Computing, 23, 4 (2005), 365-375.

[Liu et al 1998] Liu, H.C., Hong, T.S., Herman, M., Camus, T., Chellappa, R.: "Accuracy vs Efficiency Trade-offs in Optical Flow Algorithms"; Computer Vision and Image Understanding, 72, 3 (1998), 271-286.

[Lucas and Kanade 1984] Lucas, B.D., Kanade, T.: "An Iterative Image Registration Technique with an Application to Stereo Vision"; In Proc. of the DARPA Image Understanding Workshop, (1984), 121-130.

[McCane et al 2001] McCane, B., Novins, K., Crannitch, D., Galvin, B.: “On Benchmarking Optical Flow”; Computer Vision and Image Understanding, 84, 1 (2001), 126–143.

[Mesbah 99] Mesbah, M.: “Gradient-based optical flow: a critical review”; Proc. of the Fifth Int. Symp. on Signal Processing and Its Applications. ISSPA '99, 1, (1999), 467-470.

[Treves et al 1994] Treves, P., Konrad, J., "Motion estimation and compensation under varying illumination"; Proc. IEEE Int. Conf. on Image Processing, ICIP-94, 1, (1994), 373-377.

[VIS] Through-flow across the Yosemite Valley sequence. Vislist FTP site, ftp://ftp.vislist.com/SHAREWARE/CODE/OPTICAL-FLOW/.

[ISE] ISE Foundation software for FPGAs. Xilinx company, http://www.xilinx.com/ise/logic_design_prod/foundation.htm.

[Zabih and Woodfill 1994] Zabih, R., Woodfill, J.: "Non-parametric Local Transforms for Computing Visual Correspondence"; Lecture Notes in Computer Science, 801, (1994), 151-158.