ПРОГРЕСИВНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

ПРОГРЕССИВНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

PROGRESSIVE INFORMATION TECHNOLOGIES

UDC 004.93

Amanatidis Dimitrios[1], Dossis Michael[2], Androulidakis Iosif[3]

[1]PhD, Adjunct Assistant Professor, Department of Computer Engineering, Technological Educational Institute of Western Macedonia, Kastoria, Greece
[2]PhD, Associate Professor, Department of Computer Engineering, Technological Educational Institute of Western Macedonia, Kastoria, Greece
[3]PhD, Network Operation Centre, University of Ioannina, Ioannina, Greece

IMAGE CONTOUR SEGMENTATION IN HARDWARE

The use of Behavioural Synthesis for hardware generation of a contour-based image segmentation method is considered. The segmentation method chosen, is a well-known, state-of-the-art, robust, efficient and fast-converging one, that combines functionals depending on the curve geometry and image properties in a level-set framework. The cost function sought to be minimized, is formulated as a weighted sum of three integral measures; a robust alignment term that leads the evolving surface to the edges of the desired object, a minimal variance term that measures the homogeneity inside and outside the object, and a geodesic active surface term that is used mainly for regularization. The algorithm is initially implemented in MatLab and ADA and subsequently, it is ported to our Behavioural Synthesis tool, the CCC HLS framework, which is capable of delivering correct-by-construction RTL VHDL implementations of computation-intensive applications. This way, behavioural ADA specifications are transformed into RTL micro-architectures which then can be easily implemented by commercial RTL synthesizers.

Keywords: Image Contour Segmentation, High Level Synthesis, Custom Coprocessors Compilation, FPGA Implementation.

NOMENCLATURE

CCC Custom Coprocessors Compilation;
HLS High Level Synthesis;
RTL Register Transfer Level;
ITF Intermediate Tables Format;
PARCS Parallel Abstract Resource – Constrained Scheduling.

INTRODUCTION

Lately, there has been a substantial progress in partial differential equations and variational approaches in various colour image processing tasks. Among the associated computer vision application, one may typically find on one hand; applications aiming to remodel (in the PDE/Variational framework) more traditional tasks or on the other hand some emerging applications [1]:

– Image restoration, which is historically considered to be one of the oldest aims. Moreover, improving the quality of the image is quite frequently, one of the first, necessary preprocessing steps taken.

– Image segmentation, one of the most important steps in image analysis, with its own well established theoretical objectives and methodologies.

– Image and video inpainting, used for restoration of photograph cracks or errors due to image transmission and image compression and coding applications.

– Image decomposition, into a sum of a geometric part and an oscillating patterns (texture) part, used mainly in image compression.

– Image classification, where variational models are introduced in the place of more well-studied approaches such as region growing and stochastic (mainly Markov Random Field) based ones.

Accurate segmentation of various types of imagery, is a well-studied, non-trivial, image and application dependent task, which is however an essential step towards higher level image understanding. It combines the early vision preprocessing stages where salient features are highlighted while others suppressed, and allows us to move to a more effective scene analysis stage. Applications of image segmentation can be found in a broad range of disciplines; from medical diagnostic applications and inspection of manufactured products to military, security and automotive industry applications; traffic control systems and video surveillance to face, fingerprint and iris recognition to name a few.

© Amanatidis Dimitrios, Dossis Michael, Androulidakis Iosif, 2015
DOI 10.15588/1607-3274-2015-4-10

Generally speaking, image segmentation techniques can be either region-based or edge-based, which take into consideration the basic concepts of similarity and discontinuity respectively. In many cases, segmentation is formulated as an optimization problem where a set of unknown parameters have to be estimated. With the level-set approach the problem is posed as that of tracking a moving interface. Curve and surface evolution can be computed without having to parameterize the objects (Eulerian rather than Lagrangian approach) which can undergo complex topological deformations such as merging, splitting and developing holes.

The aim of this work is to present the implementation of a computationally demanding, image contour segmentation algorithm in hardware. In essence, the algorithm has to be discretized so that it can be efficiently ported to our CCC HLS framework and thereby serve as a paradigm for future implementations. Section I gives a brief problem statement.

Section II reviews related work. Section III outlines the theory of geometric active contours and Section IV presents the experimental results. Section V stresses the relation of verification and synthesis flows for our experiments and the last section discusses and concludes our work.

1 PROBLEM STATEMENT

The method's input data as discussed also in later sections are: the grayscale image *I*, the initial contour φ_{initial}, the three weighting coefficients (w_1, w_2, w_3) for the respective contributing terms in the optimization scheme, the time step τ, and the loop termination criteria; *delta* for convergence and *iter* for the maximum number of iterations. The method's output is the final contour, φ_{final}.

2 LITERATURE REVIEW

The desired properties of a good image segmentation method, as defined in [2], are that it produces homogeneous, in a sense, regions, with a simple hole-less interior, clearly distinct from their adjacent regions, having accurate and simple borders. There are mainly two broad segmentation strategies which can be seen as being one the «dual» of the other: The first one mainly exploits the notion of homogeneity in regions, separated by sharp boundaries, to identify meaningful objects. A well-known representative method of the first approach is the seminal work by Mumford and Shah [3]. In this paper, we concentrate on the second approach which aims to segment an image by detecting the contours of the different image objects. This is actually an edge detection method, with the main principle of matching deformable curves to the contour objects by means of a suitable energy functional minimization. As examined below, various improvements have been suggested over the years to the original model leading to geodesic active contours and the level sets method.

Usually, edge detection requires differentiation to detect gray-level changes and smoothing, for noise reduction of the image. The most common method of (first-order) differentiation is the gradient, while on the other hand, smoothing typically involves filtering with a 2D Gaussian.

Quite frequently, the combined smoothing and differentiation of the image is implemented by filtering the image with the differentiated smoothing filter, as in [4] for example, where edges are defined as the zero crossing curves of the Laplacian of Gaussian (LoG) applied to the image.

Active contour models, (or deformable models) which are defined as energy-minimizing splines, with energy local minima corresponding to the desired image characteristics, started with the classical snakes [5], followed by non-variational geometric active contours [6–7] and geodesic active contours [8].

3 GEOMETRIC ACTIVE CONTOURS

In two dimensions, a simple curve defines the object boundaries. A given initial curve can evolve according to its geometry and the information in the image. The evolution is a result of minimizing an energy functional – a cost function – which is influenced by image information along the curve and the intrinsic geometry of the curve. Minimization of such a measure leads to a curve that should coincide with the boundary of the object. The first variation of the functional is used to evolve a given curve toward a significant local minimum of the functional, by applying a gradient descent flow.

A more recent method [9], examined geometric functionals that do not depend on the internal parameterization of the curve, but rather on its geometry and the image properties (geometric active contours). A weighted sum of three integral measures is used, a robust alignment term that leads the evolving surface to the edges of the desired object, a minimal variance term that measures the homogeneity inside and outside the object, and a geodesic active surface term that is used mainly for regularization. The method has also been used for segmentation of thin structures in volumetric medical images [10], where the respective weights were modified for different types of images (brain CTA, lung CT, MRI etc.).

The first functional considered is the Robust Alignment Term:

$$E_{AR}(C) = \int_0^L \left| \left\langle \nabla I(x(s), y(s)) n(s) \right\rangle \right| ds, \tag{1}$$

where the inner product gets high values if the curve normal *n* aligns with the image gradient direction, therefore we seek to maximize it.

The second functional is based on the Robust Minimal Variance criterion (proposed in [11]) which is given by:

$$E_{RMV}(C) = \iint_{\Omega c} \left| I(x,y) - c_1 \right| dxdy + \iint_{\Omega/\Omega c} \left| I(x,y) - c_2 \right| dxdy, \tag{2}$$

where c_1 and c_2 are the mean intensities inside and outside the contour respectively and in the optimal case we look for the best separating contour. The term is of high importance in noisy images.

Finally, the Geodesic Active Contour functional which is sought to be minimized (as it is an inverse edge indicator) is given by:

$$E_{GAC}(C) = \int_0^L g(C(s)) ds. \tag{3}$$

This regularization term can be particularly useful in order to control the other two contributing terms (robust alignment and minimal variance).

Now, computing the first variations for each of the previous functionals, the optimization problem is posed as:

$$\operatorname{argmax}_{C, c_1, c_2} E(C, c_1, c_2), \tag{4}$$

where the combined functional is given below and α, β are positive weighting constants, chosen empirically depending on the image, with α usually much smaller than β. A suggested rule of thumb for determining the best coefficients is that, when the image has a large amount of noise, β should be large, else it should be small. Moreover, when the variance of gray scales inside the object is large, should be small:

$$E(C,c_1,c_2) = E_{\mathrm{AR}}(C,c_1,c_2) - \alpha E_{GAC}(C) - \beta E_{MV}(C)\,, \quad (5)$$

which has the following Osher-Sethian [12] level-set formulation:

$$\varphi_t = \begin{bmatrix} sign(\langle \nabla\varphi, \nabla I\rangle)\Delta I + \alpha div(g(x,y)\frac{\nabla\varphi}{|\nabla\varphi|}+ \\ \beta(c_2-c_1)(I-\frac{c_1+c_2}{2}) \end{bmatrix} |\nabla\varphi|\,. \quad (6)$$

The above formulation is shown in [9] to be solved numerically using a locally one-dimensional, fully implicit scheme:

$$\varphi^{n+1} = \prod_{k=1}^{2} (\mathbf{I} - \tau\alpha A_k)^{-1} (\varphi^n + \tau\eta(\varphi^n, \nabla I))\,, \quad (7)$$

where $\mathbf{I}$ is the identity matrix and the elements of the operators $A1$ and $A2$ are (η is the distance between neighbouring pixels):

$$\alpha_{ij} = \begin{cases} \frac{g_i+g_j}{2h^2}, & \text{if } j \in N(i); \\ -\sum_{k\in N(i)} \frac{g_i+g_j}{2h^2}, & if\ j=i; \\ 0, & \text{else}, \end{cases} \quad (8)$$

and the function η is:

$$\eta(\varphi,\nabla I) = sign(\langle\nabla\varphi,\nabla I\rangle)\Delta I + \beta(c_2-c_1)\left(I - \frac{c_1+c_2}{2}\right). \quad (9)$$

4 EXPERIMENTS

In this work, we expand on results that were reported earlier in [13]. This contour-based method, has been shown to be quite accurate and fast converging, in fact in all of the images that we have experimented with, a total number of 10 iterations is sufficient for convergence, i.e. increasing the number of iterations has no effect on the final segmented mask. Fig. 1 depicts the contours for each one of the first nine iterations, superimposed on the original image («mri»).

Specifically, in this section we present our experimental results and the qualitative comparison of the three implemented algorithms; the original MatLab version, a «flat» version of the original method in MatLab again and the GNU ADA «flat» version which is the input to our CCC HLS tools (Fig. 2).

In building up our framework towards an efficient hardware representation of the method, we re-implemented the algorithm in MatLab (constructing a «flat» version of the original method, i.e. with no function calls, for hardware implementation efficiency), using 32-bit wide integers only (to avoid overflowing), leaving all MatLab parallel constructs out (and merging loops where applicable),

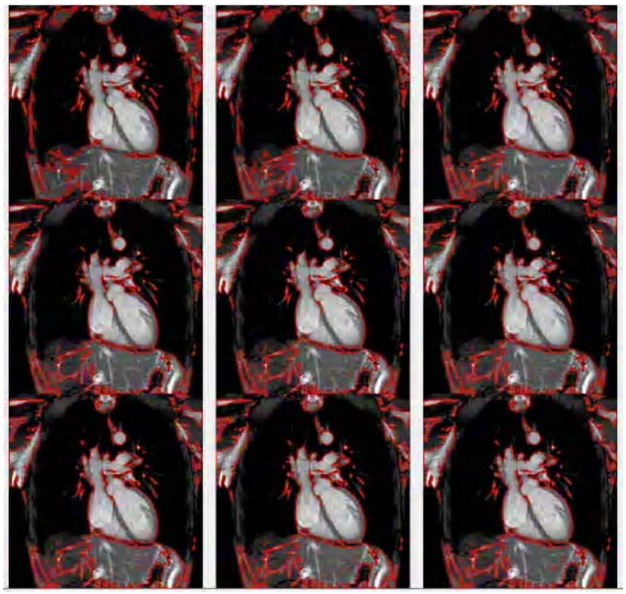

Figure 1 – Contours for the first nine iterations

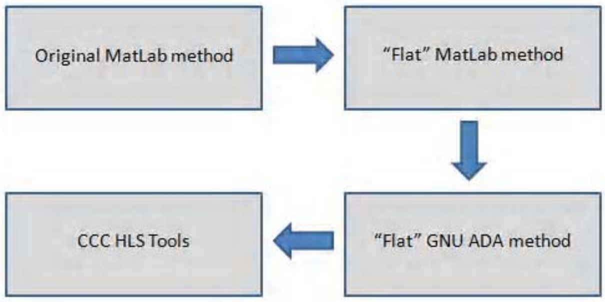

Figure 2 – The experimental framework

approximating square roots with the shifting n-th root algorithm and scaling ($X100$) some constants to avoid using any floats. Additionally, the time step τ was set to 1 rather than 0.5 and the well-studied example of $G(x, y)=1$ was used as the inverse edge indicator. This lead to a great amount of simplifications in computation, essential if the algorithm is to be implemented in hardware. The results were compared with the results obtained by the original method and were found of equivalent quality.

Fig. 3 depicts results for the «football» image with the original MatLab method. Simpler case images, such as those with no noise, uniform background and no illumination have been considered as well (as in [13]) but the main sequence tested extensively has been «football», a rather challenging one. The final contour is superimposed on the input image frame and the originally suggested parameters were left intact:

– $w1$=0.01 (weight for the geodesic active contour term);
– $w2$=4.0 (weight for the alignment forces term);
– $w3$=5.0 (weight for the minimal variance term);
– $\tau = 0.5$ (the time step);
– $iter$=30 (maximum number of iterations);
– $delta$=0.0001 (the convergence criterion threshold).

A segmented mask can be constructed by means of the final contour array elements' signs image. Note that no optimization on the contributing terms' weights has been exercised in this example.

Figure 3 – Original MatLab method results

Figure 4 – «Flat» method MatLab results

Fig. 4 displays the results with the «flat» version of the original method in MatLab with the final contour appearing in red and parameter values set to: $w1=1$, $w2=400$, $w3=500$, $\tau = 1$, $iter=10$ and $delta=1$.

The method reads the required grayscale or binary images (initial contour and input images) and any required weights and constants. Images are read and written in *.ppm* format (portable pixel maps), which makes colour image processing also feasible. An outline of the algorithm steps is following:

– read input image (I) and initial contour (φ) arrays;

– initialize some essential parameters, namely the three functional weights and the maximum number of iterations and the convergence threshold which are used as possible termination criteria;

– pre-compute first and second order derivative arrays;

– start the main while loop which tests for either convergence (using the $L2$ norm) or exceeding the maximum number of iterations – on each loop iteration do:

a) set the previous contour to be the current contour;

b) compute the minimal variance term (Fig. 5);

c) compute the robust alignment term using the derivative images (Fig. 6);

d) compute the new contour array by combining the three functional terms (Fig. 6);

e) implement the main numerical formulation scheme (previous implicit function) which utilizes Thomas algorithm (simplified significantly with the previous assumptions) for inverting the tri-diagonal operators $A1$ and $A2$;

f) set φ as a distance map of its zero set (previous redistance function) and compute the new contour array using fast–marching (previous update function);

g) check for convergence and increment iteration counter;

– end the main while loop.

Subsequently, the algorithm was implemented in GNU ADA and a comparison of the results showed equivalence in quality once more (Fig. 6). At this stage, we had to deal with some additional incompatibilities with respect to integer arithmetic, as MatLab rounds towards positive and negative infinities, whereas GNU ADA (GNAT Programming Studio) truncates towards zero.

```
--and compute Threshold (LLoydMax)
--------------------------------------------------------------------
for i in 1..HEIGHT loop
   for j in 1..WIDTH loop
      oldCout(i)(j) := Cout(i)(j);
      temp1 := Cout(i)(j);
      if temp1 < 0 then A1(i)(j) := 1;     --create A1 := mask_in
      else A1(i)(j) := 0;
      end if;
      A2(i)(j) := 1 - A1(i)(j);            --create A2 := mask_out
   end loop;
end loop;
temp1 := 0;                   --to create mask_in (A1) sum
temp2 := 0;                   --to create mask_out (A2) sum
temp3 := 0;                   --to create mask_in*I sum
temp4 := 0;                   --to create mask_out*I sum
for i in 1..HEIGHT loop
   for j in 1..WIDTH loop
      temp1 := temp1 + A1(i)(j);
      temp2 := temp2 + A2(i)(j);
      temp := A1(i)(j) * Iin(i)(j);
      temp3 := temp3 + temp;
      temp := A2(i)(j) * Iin(i)(j);
      temp4 := temp4 + temp;
   end loop;
end loop;
temp1 := temp3 / temp1;             --this is I_in
temp2 := temp4 / temp2;             --this is I_out
temp3 := (temp1 + temp2) / 2;       --this is c1
temp4 := temp2 - temp1;             --this is c2
for i in 1..HEIGHT loop
   for j in 1..WIDTH loop
      temp1 := Iin(i)(j) - temp3;
      Threshold(i)(j) := temp4 * temp1;     --force:=temp4.*(I-temp3)
   end loop;
```

Figure 5 – Computing minimal variance term

5 RESULTS

As stated, the designs were verified rapidly at the MatLab and compiled ADA code level. Moreover, RTL-level simulations were executed to prove the argument of the correctness at the level of the automatically generated RTL VHDL implementations, by the CCC behavioural synthesis tools. Thus, we ported the code to our tool, the CCC HLS framework in order to deliver correct-by-construction, Register Transfer Level (RTL), VHDL implementations of this computation-intensive application. The CCC framework consists of the frontend and the backend compiler, which communicate with each other via the ITF database.

The frontend compiler was built using compiler-compiler techniques and the backend compiler using logic

```
      end loop;
      temp := CoutT(WIDTH)(j) - CoutT(W_1)(j);
      T2(j)(WIDTH) := temp / 2;
   end loop;
   --------------------------------------------------
   --compute Alignment (Laplacian term)
   --use computed P,Q,D2I here
   --------------------------------------------------
   for i in 1..HEIGHT loop
      for j in 1..WIDTH loop
         temp1 := P(i)(j) * T1(i)(j);
         temp2 := Q(i)(j) * T2(i)(j);
         temp3 := temp1 + temp2;
         if temp3 > 0 then A1(i)(j) := -1; else
            if temp3 < 0 then A1(i)(j) := 1; else
               A1(i)(j) := 0;
            end if;
         end if;
         Alignment(i)(j) := A1(i)(j) * D2I(i)(j);
      end loop;
   end loop;
   --------------------------------------------------
   --compute new Phi (G and k are 1)
   --------------------------------------------------
   for i in 1..HEIGHT loop
      for j in 1..WIDTH loop
         temp2 := Align * Alignment(i)(j);
         temp3 := Max_Lloyd * Threshold(i)(j);
         temp4 := Balloon + temp2 + temp3 + Cout(i)(j);
         Cout(i)(j) := temp4;
      end loop;
```

Figure 6 – Final mask for GNU ADA method

programming techniques [14]. Moreover the ITF syntax and semantics are formally defined in [15]. Both of these methodologies are already patented with international patents [16]. The CCC synthesis flow is programmer-friendly, rapid and formal, which guarantees the correctness of the generated RTL implementation. Moreover, there are a number of custom options that can be used to drive the CCC compiler with specific environmental and other parameters.

The backend synthesis is optimized with the PARCS scheduler. PARCS is a formal optimizer which attempts to parallelize as many operations in the same clock cycle as possible, as long as control/data dependencies and resource constraints are obeyed.

This framework leads to behavioural ADA specifications being transformed into RTL micro-architectures, which can subsequently be implemented easily by commercial RTL synthesizers. RTL-level simulations were carried out to verify rapidly our designs and prove for correctness at the level of the automatically generated RTL VHDL implementations. Detailed experimentation with different images, validates the robustness of the proposed framework.

In order to verify the correctness of the translation a commercial tool from Mentor Graphics, Modelsim, was used to create and execute the required testbench. Test vectors (initial contour and input images) were created and fed into the algorithm (Model Under Test or MUT) in both the MatLab and the ADA environment. The actual output was finally compared to the desired one (Golden Model) to indicate a pass or fail outcome automatically.

CONCLUSION

The benefits of the proposed framework are significant. The method that was considered can efficiently detect object contours by considering the weighted sum of three functionals; robust alignment functional, motivated by the fact that in many cases the gradient direction is a good estimator of the contour orientation; minimal variance functional, which seeks the best interior-exterior separation based on mean intensity values; geodesic active contour functional, a regularization term for other dominant terms. The first variations of the three functionals are extracted, formulated in a level-set framework and solved numerically. Even though the whole process may seem quite complicated and demanding at first sight, its FPGA implementation proved feasible.

In our effort to port high level ADA coding to RTL hardware, the use of our CCC tools proved invaluable; behavioural synthesis was automatic, very fast and correct-by-construction. Future work in this area includes experimentation with other computer vision algorithms such as optical flow and graph cuts, various neural network structures, e.g. PCNN (Pulse-Coupled Neural Networks) and RBFN (Radial Basis Function Networks), or other machine learning algorithms.

REFERENCES

1. Aubert G. Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations / G. Aubert and P. Kornprobst // Applied Mathematical Sciences. – Springer Science and Business Media, LLC, 2nd ed. – 2006. –Vol. 147.
2. Haralic R. Computer and Robot Vision / R. Haralic and L. Shapiro, Eds. – 1993, chapter 16, Addison-Wesley.
3. Mumford D. Boundary detection by minimizing functionals / D. Mumford, J. Shah // Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. – San Francisco, CA, June 1985. – P. 22–26.
4. Marr D. Theory of edge detection / D. Marr, E. Hildreth // Proc of the Royal Society. – London B, 207. –1980. – P. 187–217.
5. Kass M. Snakes: Active contour models / M. Kass, A. Witkin, D. Terzopoulos // Int. Journal of Computer Vision. – 1988. – Vol. 1. – P. 321–331.
6. Malladi R. Shape modeling with front propagation: A level set approach // IEEE Transactions Pattern Analysis and Machine Intelligence. – 1995. – vol. 17. – P. 158 –175.
7. Caselles V. A geometric model for active contours in image processing / V. Caselles, F. Catte, T. Coll, F. Dibos // Numer. Math. – 1993. – Vol. 66. – P. 1–31.
8. Caselles V. On geodesic active contours / V. Caselles, R. Kimmel and G. Sapiro // Int. Journal of Computer Vision. – 1997. – Vol. 22, No. 1. – P. 61–79.
9. Kimmel R. Fast edge integration // Geometric Level Set Methods in Imaging, Vision and Graphics / R. Kimmel, S. Osher and N. Paragios, Eds. – New York : Springer-Verlag, 2003.
10. Segmentation of Thin Structures in Volumetric Medical Images / M. Holtzman-Gazit, R. Kimmel, N. Peled and D. Goldsher // IEEE Transactions On Image Processing. – 2006. –Vol. 15, No. 2.
11. Chan T. Active Contours Without Edges / T. Chan and L. Vese / / IEEE Transactions on Image Processing. – 2001. – Vol. 10, No. 2. – P. 266–277.
12. Osher S. Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations / S. Osher, J. A. Sethian // Journal of Computational Physics. – 1988. – Vol. 79 – P. 12–49.
13. Dossis M. Using Behavioural Synthesis for hardware generation of a contour-based image segmentation algorithm / M. Dossis, D. Amanatidis, I. Androulidakis // Proceedings of the International Conference on Information and Digital Technologies (IDT). – Zilina, Slovakia, July 7–9, 2015.

14. Dossis M. A Formal Design Framework to Generate Coprocessors with Implementation Options / M. F. Dossis // International Journal of Research and Reviews in Computer Science (IJRRCS). – Science Academy Publisher, United Kingdom. – August 2011. – Vol. 2, No 4. – P. 929–936. ISSN: 2079–2557, www.sciacademypublisher.com.
15. Dossis M. Intermediate Predicate Format for Design Automation Tools / M. Dossis // Journal of Next Generation Information Technology (JNIT). – May 2010. – Vol. 1, No. 1. – P. 100–117.
16. Dossis M. Patent № 1005308, 5/10/2006 / M. Dossis // By the Greek Industrial Property Organisation, 2006.

Article was submitted 09.08.2015.
After revision 16.05.2015.

Аманатідіс Дімітріос[1], Доссіс Михаїл[2], Андроулідакіс Йосип[3]
[1]Доктор філософії, ад’юнкт, асистент кафедри кафедри комп’ютерної інженерії, Інститут технологічної освіти Західної Македонії, Касторія, Греція
[2]Доктор філософії, доцент кафедри комп’ютерної інженерії, Інститут технологічної освіти Західної Македонії, Касторія, Греція
[3]Доктор філософії, Центр управління комп’ютерними мережами, Університет Яніни, Яніна, Греція

АПАРАТНА РЕАЛІЗАЦІЯ КОНТУРНОЇ СЕГМЕНТАЦІЇ ЗОБРАЖЕНЬ

Розглянуто використання поведінкового синтезу для створення апаратного забезпечення контурної сегментації зображень. В якості методу сегментації обраний добре відомий, надійний, ефективний метод, що швидко збігається, який комбінує функціонали, залежні від геометрії кривих і властивостей зображення у множині рівнів структури. Мінімізована функція витрат формулюється як зважена сума трьох інтегральних мір: стійкого вирівнювання (прагне розвинути поверхню до країв бажаного об’єкта), мінімальної дісперсії (вимірює однорідність всередині і зовні об’єкта) і геодезично активної поверхні (використовується в основному для регуляризації). Алгоритм спочатку реалізований в Matlab і ADA, а потім, він перенесений у наш інструмент Поведінкового синтезу – середовище CCC HLS, яке здатне створювати правильно побудовані RTL VHDL реалізації додатків, що інтенсивно використовують обчислення. Таким чином, поведінкові характеристики ADA перетворюються у мікроархітектури RTL, які потім можуть бути легко реалізовані за допомогою комерційних RTL синтезаторів.

Ключові слова: контурна сегментація зображень, високорівневий синтез, збірка користувальницьких сопроцессоров, ПЛІС-реалізація.

Аманатидис Димитриос[1], Доссис Михаил[2], Андроулидакис Иосиф[3]
[1]Доктор философии, адъюнкт, ассистент кафедры кафедры компьютерной инженерии, Институт технологического образования Западной Македонии, Кастория, Греция
[2]Доктор философии, доцент кафедры компьютерной инженерии, Институт технологического образования Западной Македонии, Кастория, Греция
[3]Доктор философии, Центр управления компьютерными сетями, Университет Янины, Янина, Греция

АППАРАТНАЯ РЕАЛИЗАЦИЯ КОНТУРНОЙ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ

Рассмотрено использование поведенческого синтеза для создания аппаратного обеспечения контурной сегментации изображений. В качестве метода сегментации выбран хорошо известный, надежный, эффективный и быстро сходящийся метод, комбинирующий функционалы, зависящие от геометрии кривых и свойств изображения во множестве уровней структуры. Минимизируемая функция затрат формулируется как взвешенная сумма трех интегральных мер: устойчивого выравнивания (стремится развить поверхность к краям желаемого объекта), минимальной дисперсии (измеряет однородность внутри и снаружи объекта) и геодезически активной поверхности (используется в основном для регуляризации). Алгоритм изначально реализован в MatLab и ADA, а затем, он перенесен в наш инструмент Поведенческого синтеза – среду CCC HLS, которая способна создавать правильно построенные RTL VHDL реализации приложений, интенсивно использующих вычисления. Таким образом, поведенческие характеристики ADA преобразуются в микроархитектуры RTL, которые затем могут быть легко реализованы с помощью коммерческих RTL синтезаторов.

Ключевые слова: контурная сегментация изображений, высокоуровневый синтез, сборка пользовательских сопроцессоров, ПЛИС-реализация.

REFERENCES

1. Gilles Aubert and Pierre Kornprobst Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations, *Applied Mathematical Sciences*. Springer Science and Business Media, LLC, 2nd ed, 2006, Vol. 147.
2. Haralic R. and Shapiro L., Eds. Computer and Robot Vision. 1993, chapter 16, Addison-Wesley.
3. Mumford D. and Shah J. Boundary detection by minimizing functionals, *Proc of the IEEE Conference on Computer Vision and Pattern Recognition*. San Francisco, CA, June 1985, pp. 22–26.
4. Marr D. and Hildreth E. Theory of edge detection, *Proc of the Royal Society.* London B, 207, 1980, pp. 187–217.
5. Kass M., Witkin A. and Terzopoulos D. Snakes: Active contour models, *Int. Journal of Computer Vision*, 1988, Vol. 1, pp. 321–331.
6. Malladi R. Shape modeling with front propagation: A level set approach, *IEEE Transactions Pattern Analysis and Machine Intelligence*, 1995, Vol. 17, pp. 158–175.
7. Caselles V., Catte F., Coll T. and Dibos F. A geometric model for active contours in image processing, *Numer. Math.*, 1993, Vol. 66, pp. 1–31.
8. Caselles V., Kimmel R. and Sapiro G. On geodesic active contours, *Int. Journal of Computer Vision,* 1997, Vol. 22, No. 1, pp. 61–79.
9. Kimmel R., Osher S. and Paragios N., Eds. Fast edge integration, *Geometric Level Set Methods in Imaging*, Vision and Graphics. New York, Springer-Verlag, 2003.
10. Michal Holtzman-Gazit, Ron Kimmel, Nathan Peled and Dorith Goldsher Segmentation of Thin Structures in Volumetric Medical Images, *IEEE Transactions On Image Processing*, 2006, Vol. 15, No. 2.
11. Chan T., Vese L. Active Contours Without Edges, *IEEE Transactions on Image Processing*, 2001, Vol. 10, No. 2, pp. 266–277.
12. Stanley Osher and James A. Sethian Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations, *Journal of Computational Physics*, 1988, Vol. 79, pp. 12–49.
13. Michael Dossis, Dimitrios Amanatidis and Iosif Androulidakis, Using Behavioural Synthesis for hardware generation of a contour-based image segmentation algorithm, *Proceedings of the International Conference on Information and Digital Technologies (IDT)*. Zilina, Slovakia, July 7–9, 2015.
14. Michael F. Dossis A Formal Design Framework to Generate Coprocessors with Implementation Options, *International Journal of Research and Reviews in Computer Science (IJRRCS)*. Science Academy Publisher, United Kingdom. August 2011, Vol. 2, No 4, pp. 929–936. ISSN: 2079–2557, www.sciacademypublisher.com.
15. Michael Dossis Intermediate Predicate Format for Design Automation Tools, *Journal of Next Generation Information Technology (JNIT)*. May 2010, Vol. 1, No. 1, pp. 100–117.
16. Dossis M. Patent number 1005308, 5/10/2006 by the Greek Industrial Property Organisation, 2006.