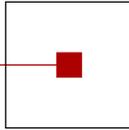


s c c h

software competence center
hagenberg



Advances in Knowledge-Based Technologies

Proceedings of the
Master and PhD Seminar
Winter term 2011/12, part 1

Softwarepark Hagenberg
SCCH, Room 0/2
24 November 2011

Software Competence Center Hagenberg
Softwarepark 21
A-4232 Hagenberg
Tel. +43 7236 3343 800
Fax +43 7236 3343 888
www.scch.at

Fuzzy Logic Laboratorium Linz
Softwarepark 21
A-4232 Hagenberg
Tel. +43 7236 3343 431
Fax +43 7236 3343 434
www.flll.jku.at

Program

Session 1. Chair: Roland Richter

- 9:00 Radko Mesiar:
Copulas and integrals
- 9:30 Martin Reinhardt:
Steerable Wavelet Frames and their Applications

Session 2. Chair: Bernhard Moser

- 10:15 Carlos Cernuda:
Genetic Operators Design for Variable Selection Using Genetic Algorithms
- 10:45 Ciprian Zavoianu:
Improving the Parsimony of Regression Models for an Enhanced Genetic Programming Process
- 11:15 Jean-Luc Bouchot:
Local Structural Similarities Based on Distance Transforms for Image Comparison

Copulas and integrals

Radko Mesiar

Faculty of Civil Engineering
Slovak University of Technology
Radlinského 11, 813 68 Bratislava, Slovakia
radko.mesiar@stuba.sk

The stochastic dependence structure of bivariate random variables captured by copulas can be applied in the integration approaches. Indeed, a copula $C : [0, 1]^2 \rightarrow [0, 1]$ can express the connection between function values and measure values through an integral $I_{C,m}$ by $I_{C,m}(a 1_A) = C(a, m(A))$, where m is the (monotone) measure and $a \in [0, 1]$ is a constant. A class of copula-based universal integrals (they are well defined for any measurable space (X, \mathcal{A}) , any normed monotone measure $m : \mathcal{A} \rightarrow [0, 1]$ and any \mathcal{A} -measurable function $f : X \rightarrow [0, 1]$) was introduced in [1] by

$$I_{C,m}(f) = P_C(\{(x, y) \in [0, 1]^2 \mid y \leq m(f \geq x)\}),$$

where P_C is the probability measure on Borel subsets of $[0, 1]^2$ induced by C . As special instances recall that $C = \Pi$ yields the Choquet integral, while $C = \text{Min}$ yields the Sugeno integral. If X is finite, $X = \{1, \dots, n\}$, and $\mathcal{A} = 2^X$, then we have two equivalent formulae exploited, e.g., in multicriteria decision support,

$$\begin{aligned} I_{C,m}(f) &= \sum_{i=1}^n (C(f_{(i)}, m(A_i)) - C(f_{(i-1)}, m(A_i))) = \\ &= \sum_{i=1}^n (C(f_{(i)}, m(A_i)) - C(f_{(i)}, m(A_{i+1}))), \end{aligned}$$

where $(f_{(1)}, \dots, f_{(n)})$ is a nondecreasing permutation of $(f(1), \dots, f(n))$, $A_{(i)} = \{j \mid f(j) \geq f_{(i)}\}$, with conventions $f_{(0)} = 0$ and $A_{n+1} = \emptyset$.

Looking on integrals $I_{C,\cdot}$ as functionals, one can introduce an axiomatic approach, too. For example, comonotone additivity is a genuine property of the Choquet integral [4], while the comonotone maxitivity and the min-homogeneity characterize the Sugeno integral [2]. We give a general axiomatic characterization of discrete copula-based universal integrals. A substantial role of the comonotone modularity will be shown, linking these integrals and OMA operators [3].

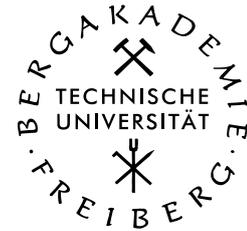
Acknowledgment. The research summarized in this paper was supported by the Grant APVV-0073-10.

References

1. Klement, E.P., Mesiar, R. and Pap, E.: A universal integral as a common frame for Choquet and Sugeno integral. IEEE Transactions on Fuzzy Systems **18**, pp. 178–187 (2010)

2. Marichal, J.-L.: An axiomatic approach of the discrete Sugeno integral as a tool to aggregate interacting criteria in a qualitative framework. *IEEE Trans. Fuzzy Syst.* **9(1)**, pp. 164 – 172 (2001)
3. Mesiar, R., Mesiarová – Zemánková, A.: The ordered modular averages. *IEEE Transactions on Fuzzy Systems* **19**, pp. 42 – 50 (2011)
4. Schmeidler, D.: Integral representation without additivity. *Proc. Amer. Math. Soc.* **97**, pp. 255–261 (1986)

Technische Universität Bergakademie Freiberg
Faculty of mathematics and informatics
Institute of applied analysis



FLLL Seminar Hagenberg

Steerable Wavelet Frames and their Applications

Martin Reinhardt

Applied Mathematics
Specialization: Communication Technologies

November 23, 2011

Tutor/First Proofreader:
Dr. Dipl.Phys.in Bettina Heise
CDL-MS-MACH Linz/ FLLL Hagenberg

Due to the fact, that analytical wavelets are widely used to decompose 1D signals, it yields to find a multidimensional analogue. With such an analogue, called monogenic wavelet, it is possible to combine the principle of multiresolution analysis of wavelet frames with the approach to decompose a multidimensional signal, e.g. an image, into amplitude, phase and phase direction. Applications can be found in the enhancement of contrast, detection of edges and filtering of details.

References

- [ACM08] Aldroubi, Akram, Cabrelli, Carlos, and Molter, Ursula M. “Wavelets on irregular Grids with arbitrary Dilation Matrices, and frame atoms for $L^2(\mathbb{R}^d)$ ”. In: *Mathematics Subject Classification* 42C40 (Feb. 2008).
- [But03] Butz, Tilmann. *Fouriertransformation für Fußgänger*. 2003. ISBN: 3-519-20202-6.
- [FS01] Felsberg, M. and Sommer, G. “The monogenic signal”. In: *IEEE Transactions on Signal Processing* 49 (Dec. 2001).
- [HSMF10] Held, Stefan, Storath, Martin, Massopust, Peter, and Forster, Brigitte. “Steerable Wavelet Frames Based on Riesz Transform”. In: *IEEE Transactions on Image Processing* 19 (Mar. 2010).
- [LMR98] Louis, Alfred Karl, Maaß, Peter, and Rieder, Andreas. *Wavelets. Theorie und Anwendungen*. Teubner Verlag, 1998. ISBN: 3-519-12094-1.
- [MW99] Müller-Wichards, Dieter. *Transformationen und Signale*. Teubner Verlag, 1999. ISBN: 3-519-02742-9.
- [USV] Unser, Michael, Sage, Daniel, and Ville, Dimitri Van De. “Multiresolution monogenic signal analysis using the Riesz-Laplace wavelet transform”.
- [UV10] Unser, Michael and Ville, Dimitri Van De. “Wavelet steerability and the higher-order Riesz transform”. In: *IEEE Transactions on Image Processing* 10 (2010).
- [Wut10] Wutzig, Michael. “Multiresolution Analysis monogener Signale im \mathbb{R}^3 ”. Diplomarbeit. TU Bergakademie Freiberg, Sept. 2010.

1 Generalization of the analytical signal

At first, the analytical signal and its multidimensional analogue, the Riesz transformation should be introduced.

Let an arbitrary real-valued function $f \in L^2(\mathbb{R}, \mathbb{R})$. Then it yields for the analytic signal $f_a \in L^2(\mathbb{R}, \mathbb{C})$, $f_a = f + i\mathcal{H}f$, where

$$\mathcal{H} : L^2(\mathbb{R}) \rightarrow L^2(\mathbb{R}) \quad (1)$$

$$\mathcal{H}f = \frac{1}{\pi} \text{P. V.} \int_{\mathbb{R}} \frac{f(x)}{x-t} dx \quad (2)$$

means the Hilbert transform (here, P. V. denotes the Cauchy principal value). the complex-valued analytical signal can now decomposed in its amplitude $a(t)$ and its phase $\varphi(t)$.

$$f_a(t) = a(t)e^{i\varphi(t)} = a(t)(\cos \varphi(t) + i \sin \varphi(t)) \quad (3)$$

Obviously, it follows $f(t) = a(t) \cos \varphi(t)$ for reconstruction. The derivative $\varphi'(t)$ is called instantaneous frequency.

For the Generalization of the Hilbert transform, the Riesz transform is used, because it is the only possibility to come along with rotations. This is very important for the construction of isotropic filters. This transformation can also easily implemented in the Fourier domain. The starting point of the generalization is the Hilbert transform in the Fourier domain: $\mathcal{H} : L^2(\mathbb{R}) \rightarrow L^2(\mathbb{R})$

$$\widehat{\mathcal{H}f}(\xi) = i \operatorname{sgn}(\xi) \widehat{f}(\xi) = i \frac{\xi}{|\xi|} \widehat{f}(\xi) \quad (4)$$

where $i^2 = -1$. The kernel (Fourier multiplier) is therefore $i \operatorname{sgn}(\xi) = i(\xi/|\xi|)$. Consider an orthonormal basis $\{e_\alpha\}_{\alpha=1}^n$ of \mathbb{R}^n . To extend the Hilbert transform to higher dimensions now, the Fourier multiplier $i(\xi_\alpha/|\xi|)$ is applied to the signal for each component $\xi_\alpha = \langle \xi, e_\alpha \rangle$, $\alpha = 1, \dots, n$ of the multidimensional signal.

So the Riesz transform $\mathcal{R} : L^2(\mathbb{R}^n) \rightarrow (L^2(\mathbb{R}^n))^n$ is defined in the Fourier domain by

$$\widehat{\mathcal{R}f} = \left(\widehat{\mathcal{R}_1 f}, \dots, \widehat{\mathcal{R}_n f} \right) \quad (5)$$

where $\mathcal{R}_\alpha : L^2(\mathbb{R}^n) \rightarrow L^2(\mathbb{R}^n)$ is the α th component

$$\widehat{\mathcal{R}_\alpha f}(\xi) = i \frac{\xi_\alpha}{\|\xi\|} \widehat{f}(\xi), \quad \alpha = 1, \dots, n \quad (6)$$

By using the orthonormal base $\{e_\alpha\}_{\alpha=1}^n$, the Riesz transform can be written in the form

$$\mathcal{R} = \sum_{\alpha=1}^n e_\alpha \mathcal{R}_\alpha$$

The next point is to define all important operators for dilation (D_l), translation (T_b) und rotation (ρ) in the usual way:

$$D_l, T_b, \rho: L^2(\mathbb{R}^n) \rightarrow L^2(\mathbb{R}^n)$$

$$D_l f(x) = l^{n/2} f(lx), \quad l \in \mathbb{R}^+$$

$$T_b f(x) = f(x - b), \quad b \in \mathbb{R}^n$$

$$\rho f(x) = f(\rho^{-1}x), \quad \rho = \{\rho_{\alpha,\beta}\}_{\alpha,\beta=1}^n \in O(\mathbb{R}^n)$$

The following properties characterizing the Riesz and the Hilbert transform uniquely.

$$T_b \mathcal{H} = \mathcal{H} T_b \qquad T_b \mathcal{R}_\alpha = \mathcal{R}_\alpha T_b \qquad (7)$$

$$D_l \mathcal{H} = \mathcal{H} D_l \qquad D_l \mathcal{R}_\alpha = \mathcal{R}_\alpha D_l \qquad (8)$$

$$\|\mathcal{H}\|_{op} \leq 1 \qquad \|\mathcal{R}_\alpha\|_{op} \leq 1 \qquad (9)$$

$$\mathcal{H} \circ \rho = \rho_{1,1}(\rho \circ \mathcal{H}) \qquad \mathcal{R}_\alpha \circ \rho = \sum_{\beta=1}^n \rho_{\alpha,\beta}(\rho \circ \mathcal{R}_\beta) \qquad (10)$$

These features are showing, that the Riesz transform has got all relevant properties of the Hilbert transform, which supports the generalization. Equations (7) and (8) are describing the wavelet generating operations. With the equation (10) the Riesz transformed wavelets can be steered into all directions with the unit vector $a \in \mathbb{R}^n$, $\|a\| = 1$ and $\rho \in O(n)$ with $a = \rho e_1$.

$$\mathcal{R}_\alpha = \rho^{-1} \mathcal{R}_1 \rho = \sum_{\beta=1}^n \rho_{1,\beta} \mathcal{R}_\beta \qquad (11)$$

Especially this property is the fundamental of the easy applications in image processing because of the orthogonal directions of datasets. The main definition of the monogenic signal can be found in [FS01]:

Let $f \in L^2(\mathbb{R}^n, \mathbb{R})$ be a real-valued function. Let $e_0 \in \mathbb{R}^{n+1}$ be a vector that extends the ONB $\{e_\alpha\}_{\alpha=1}^n$ of \mathbb{R}^n of \mathbb{R}^{n+1} . The monogenic signal f_m is defined as

$$f_m = e_0 f + \mathcal{R} f = e_0 f + \sum_{\alpha=1}^n e_\alpha \mathcal{R}_\alpha f \qquad (12)$$

In the notation of (5)

$$f_m = (f, \mathcal{R}_1 f, \dots, \mathcal{R}_n f) \qquad (13)$$

Therefore $f_m: \mathbb{R}^n \rightarrow \mathbb{R}^{n+1}$ is a vector-valued mapping and because of the bounding property in (9) element of $L^2(\mathbb{R}^n, \mathbb{R}^{n+1})$.

2 A fitting exponential function

Similar to the decomposition of the analytic signal (3) into phase and amplitude components it is aimed to find such a decomposition for the monogenic signal. This decomposition consists of two main functions $a: \mathbb{R} \rightarrow \mathbb{R}$ and $\exp(i\varphi(\cdot)): \mathbb{R} \rightarrow \mathbb{C}$. On the image \mathbb{R}^{n+1} there isn't such a *a priori* algebraic structure to define a vector-valued multiplication. With the help of the underlying ONB and following definitions

$$\begin{aligned} e_\alpha e_\beta &= -e_\beta e_\alpha \\ e_\alpha e_0 &= e_0 e_\alpha \\ e_\alpha^2 &= -e_0 \\ e_0^2 &= e_0 \end{aligned}$$

for all $\alpha \neq \beta$, $\alpha, \beta = 1, \dots, n$, an extension of the multiplication to all elements $x, y \in \mathbb{R}^{n+1}$ can be found via linear representation with respect to the ONB:

$$xy = \left(\sum_{\alpha=0}^n e_\alpha x_\alpha \right) \left(\sum_{\beta=0}^n e_\beta y_\beta \right) = \sum_{\alpha=0}^n \sum_{\beta=0}^n e_\alpha e_\beta x_\alpha y_\beta \quad (14)$$

This in general non-commutative vector-multiplication extends the \mathbb{R}^{n+1} to a Clifford-Algebra \mathbb{R}_n of the dimension 2^n . (More information about this scheme of extension of a multiplication can be found in [HSMF10]) With the identity $e_0 \leftrightarrow 1$, x has a representation of a pseudo-vector:

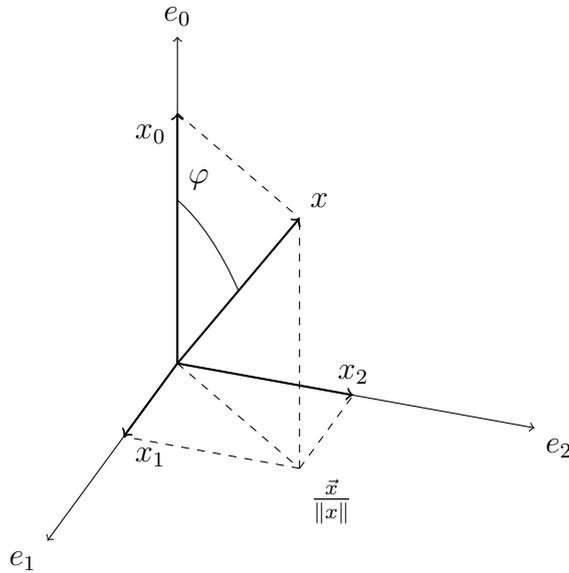
$$x = x_0 + \vec{x}, \text{ wobei } \vec{x} = \sum_{\alpha=1}^n e_\alpha x_\alpha \quad (15)$$

If φ denotes the angle between the scalar and the vector part then it follows from $\tan \varphi = \|\vec{x}\|/x_0$

$$\begin{aligned} x &= \|x\| \left(\frac{x_0}{\|x\|} + \frac{\vec{x}}{\|\vec{x}\|} \cdot \frac{\|\vec{x}\|}{\|x\|} \right) \\ &= \|x\| \left(\cos \varphi + \frac{\vec{x}}{\|\vec{x}\|} \sin \varphi \right) \end{aligned}$$

Description:

- $\|x\|$ Amplitude
- φ Phase
- $\vec{x}/\|\vec{x}\|$ Phasenrichtung von x



The last thing, needed for a full decomposition like in (3), is the exponential function in \mathbb{R}^{n+1} . For the usual exponential series it is important, that the subspace \mathbb{R}^{n+1} of the algebra \mathbb{R}_n is invariant under power operations (proof in [HSMF10]). Here, a small example should be enough:

$$x^2 = x_0^2 + 2\vec{x}x_0 + \vec{x}\vec{x} = (x_0^2 - \|\vec{x}\|^2) + 2\vec{x}x_0 \in \mathbb{R}e_0 + \mathbb{R}\vec{x}$$

Due this fact, an exponential function can be found with

$$\exp: \mathbb{R}^{n+1} \rightarrow \mathbb{R}^{n+1}, x \rightarrow \sum_{n \in \mathbb{N}} \frac{x^n}{n!}$$

It holds that $\exp(x_0 + \vec{x}) = (x_0) \exp(\vec{x})$ for all $x \in \mathbb{R}^{n+1}$. Furthermore, note that

$(\vec{x}/\|\vec{x}\|)^2 = -1$ With $\varphi \in \mathbb{R}$ and

$$\begin{aligned}
 \exp\left(\frac{\vec{x}}{\|\vec{x}\|}\varphi\right) &= \sum_{k=0}^{\infty} \frac{\left(\frac{\vec{x}}{\|\vec{x}\|}\varphi\right)^k}{k!} \\
 &= \sum_{k=0}^{\infty} \left(\frac{\vec{x}}{\|\vec{x}\|}\right)^{2k} \frac{\varphi^{2k}}{(2k)!} + \sum_{k=0}^{\infty} \left(\frac{\vec{x}}{\|\vec{x}\|}\right)^{2k+1} \frac{\varphi^{2k+1}}{(2k+1)!} \\
 &= \sum_{k=0}^{\infty} (-1)^k \frac{\varphi^{2k}}{(2k)!} + \frac{\vec{x}}{\|\vec{x}\|} \sum_{k=0}^{\infty} (-1)^k \frac{\varphi^{2k+1}}{(2k+1)!} \\
 &= \cos \varphi + \frac{\vec{x}}{\|\vec{x}\|} \sin \varphi
 \end{aligned}$$

it can be shown that

$$x = \|x\| \exp\left(\frac{\vec{x}}{\|\vec{x}\|}\varphi\right) \quad (16)$$

3 Implementation and application aspects

With an arbitrary isotropic wavelet frame ψ and the use of Clifford Frames (detailed explanations in [HSMF10]) the approach of multiresolution analysis and the aim of decomposition can be combined.

The most important points to know are:

- The wavelet controls the steepness, the edges are falling.
- The tightness of a wavelet frame (wavelet frames which match the Parseval's identity) is important to use the same filter bank for analysis and synthesis.
- The filters are applied in the Fourier domain, because there convolutions are just multiplications.
- The singularity in the kernel of the Riesz transform can be handled by zero-setting of the DC-Component of the picture.
- The compact support property ensure a perfect reconstruction of the picture by the Shannon sampling theorem.
- A dilation matrix $\sqrt[k]{2}I_n$ is useful to get a $(k+1)$ -channel filter bank with
 - the low-pass filters h_0
 - several band pass filters h_j , $j = 1, \dots, k-1$
 - the high-pass filter h_k

With the help of the coefficients provided by the algorithm (see in section 4) it is possible to calculate different useful properties of an image:

- Amplitude: $A_{s,j} = \sqrt{d_{s,j}^2 + (\mathcal{R}_1 d_{s,j})^2 + (\mathcal{R}_2 d_{s,j})^2}$
- Phase: $P_{s,j} = \text{atan2}(\mathcal{N}_{s,j}, d_{s,j})$
- Phase orientation: $O_{s,j} = \begin{cases} \text{atan2}\left(\frac{\mathcal{R}_2 d_{s,j}}{\mathcal{N}_{s,j}}, \frac{\mathcal{R}_1 d_{s,j}}{\mathcal{N}_{s,j}}\right), & \mathcal{R}_2 d_{s,j} \geq 0 \\ \text{atan2}\left(\frac{\mathcal{R}_2 d_{s,j}}{\mathcal{N}_{s,j}}, \frac{\mathcal{R}_1 d_{s,j}}{\mathcal{N}_{s,j}}\right) + \pi, & \text{sonst} \end{cases}$

where $\mathcal{N}_{s,j} = \sqrt{(\mathcal{R}_1 d_{s,j})^2 + (\mathcal{R}_2 d_{s,j})^2}$. The indices s, j mean the scale-index s and the sub-channel-index j . Furthermore, a two-argument arctan is defined as usual by

$$\text{atan2}(x, y) = \arg(x + iy)$$

A reconstruction of the picture by its phase and amplitude can be performed by

$$d_{s,j} = A_{s,j} \cos(P_{s,j})$$

The main application of the combination of phase and amplitude decomposition is the extraction of image-features on different detail levels. Another important application is the 'equalisation of brightness'. The aim of this is to reconstruct a corrupted image only by its phase. The amplitude is replaced by a default noise-term:

$$\tilde{d}_{s,j} = \cos(P_{s,j})$$

4 functionplots and images



Fig. 1: Originalbild von Barbara

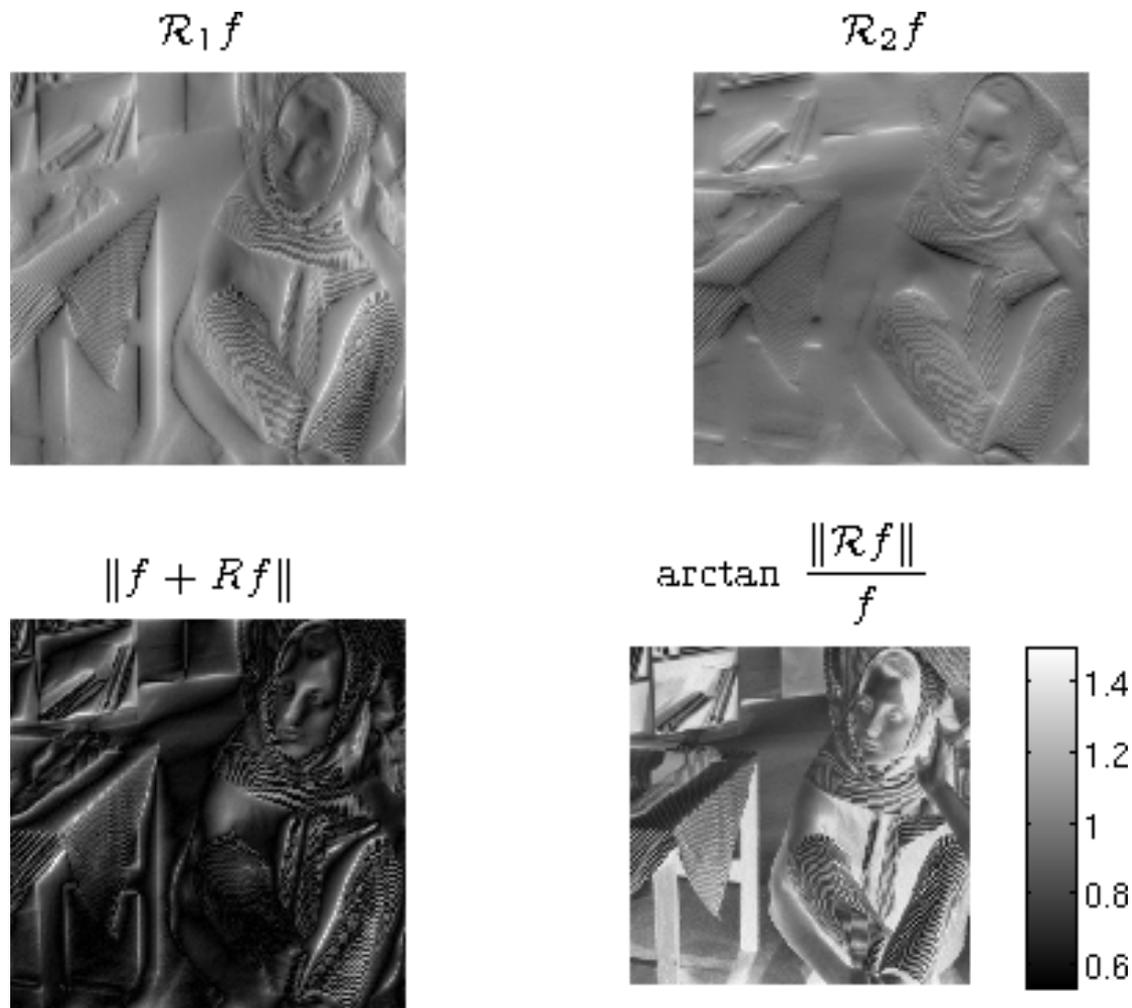


Fig. 2: Resultate der Transformationen von Barbara

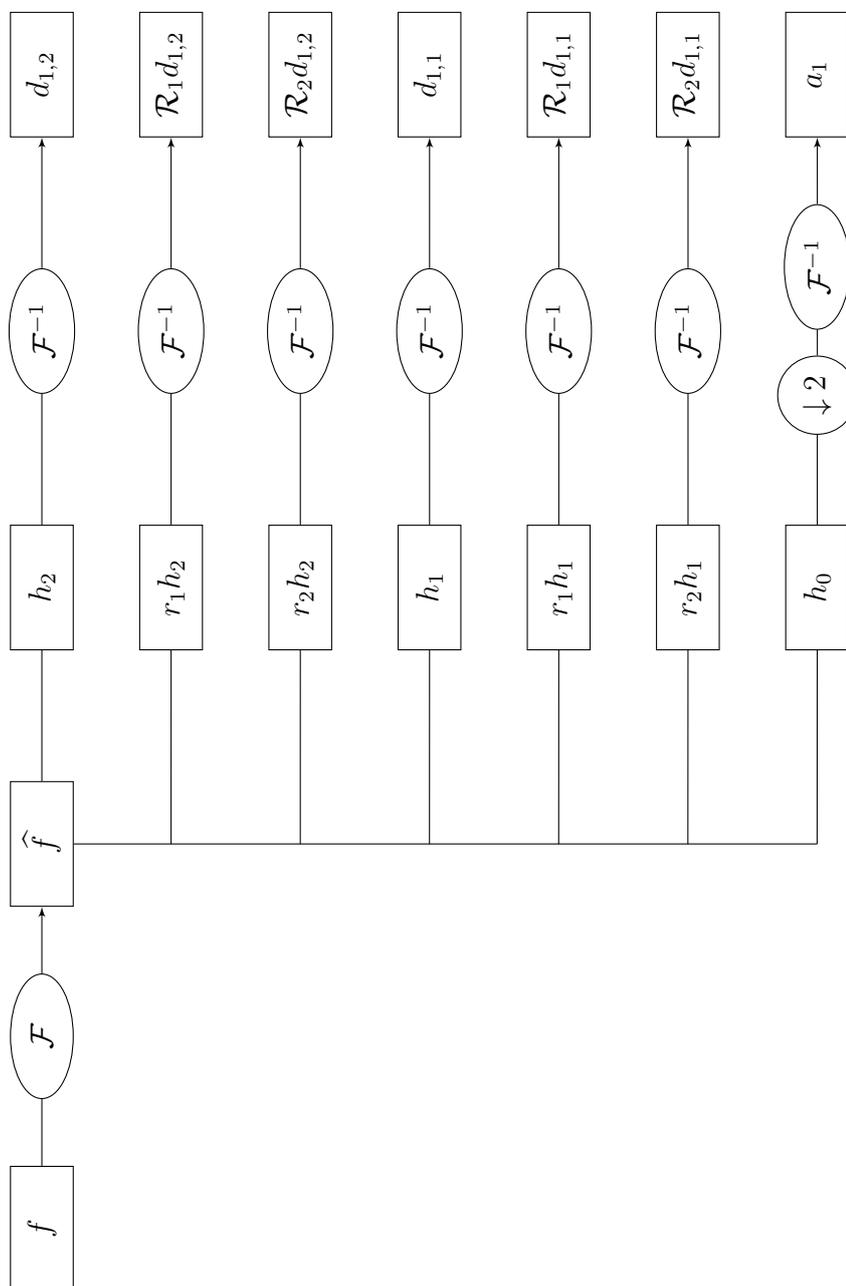


Fig. 3: Analyse mit Hilfe der Filterbank am Beispiel $k = 2$

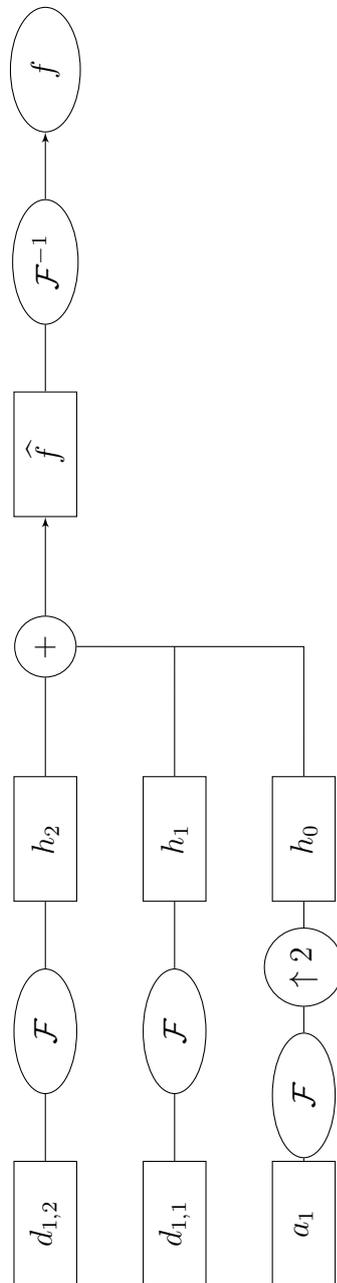


Fig. 4: Synthese mit Hilfe der Filterbank am Beispiel $k = 2$

Genetic Operators Design for Variable Selection using Genetic Algorithms

AdvKBT Winter 2011/12, 1

Carlos Cernuda^a

^a*Fuzzy Logic Lab Linz (FLLL)*

*Department of Knowledge-Based Mathematical Systems
Johannes Kepler University Linz, Austria*

Abstract

Nowadays the techniques employed in data acquisition provide huge amounts of data. For instance, Near Infra-Red (NIR) spectral data consist on thousands of wavelengths (variables). Some parts of the information are related to the others, making desirable a way to reduce the number of variables, i.e. dimensionality reduction, losing as less information as possible, in order to decrease computational times and complexity when applying any Data Mining technique, e.g. for classification or regression purposes. Genetic Algorithms(GA) offer the possibility of selecting which variables contain the most relevant information in order to represent all the original ones. The traditional genetic operators seem to be too general leading to results which could be improved by means of designed genetic operators that employ some available problem specific information.

Keywords: Variable Selection, Dimensionality Reduction, Genetic Algorithm, Forward Selection, Design Genetic Operators.

1. Introduction: Motivation and Methodology

Computational studies of Darwin's Evolution and Natural Selection theories have led to numerous models to tackle optimization problems [6, 3, 16, 5]. GA's are one of

Email address: `carlos.cernuda@jku.at` (Carlos Cernuda)

the most relevant. They are parallel iterative evolution-based optimizers based on the application of selection, mutation, and recombination to a population of competing problem solutions [13, 7]. There are many ways of constructing GA's. In general every GA must have a fitness function, in order to have the possibility of determining the goodness of an individual, and genetic operators, to simulate natural evolution of the population of individuals, i.e. solutions of the optimization problem. Usual genetic operators suffer from a lack of exploitation capacity of the existing problem specific information. Here is where clever design strategies can lead to appropriate operators which allow the GA to converge faster to high quality solutions.

This paper is structured as follows: in section 2 we introduce the usual genetic operators, describing all the relevant parts, and the description of the data set used for comparison and how that comparison is done. Section 3 describes the design original parts of our GA. In section 4 the results obtained are shown and commented. Finally, section 5 gives a general overview and sets the trend for future works.

2. Methodology and Experimental Setup

2.1. Methodology

One crucial fact when using a GA is the way to codify the individuals. In this case binary vectorial representation is the most adequate, meaning that an individual $x = (x_1, x_2, \dots, x_m)$ contains values $x_i = 1$ if the i -th variable is considered in the model or $x_i = 0$ if not. Not less important is the fitness function, in terms of which we decide how good an individual is. In the case we will consider here, regression models, the usual way is build a fast model and calculate the prediction error. Thus the lower the error the better the individual. An initial population is needed. There are several ways of generating an initial population, being random generation the most popular. Given a pair of solutions (parents) we generate a new individual (offspring) by means of a crossover operator. In order to favour diversity we will give the offspring the chance of undergoing slightly, i.e. mutation operator. In principle not all individuals must be employed as parents, just some selected ones according to some criteria, i.e. selection operator.

2.2. Data set description

The KCCS data set consists on 25 multiple instances, i.e. every instance is a set of 20 consecutive measurements, corresponding to 25 concrete moments. The type of data is NIR spectral data, considering 982 equidistant wavelengths. Therefore we are handling a matrix with 25 blocks of 20 rows and 982 columns.

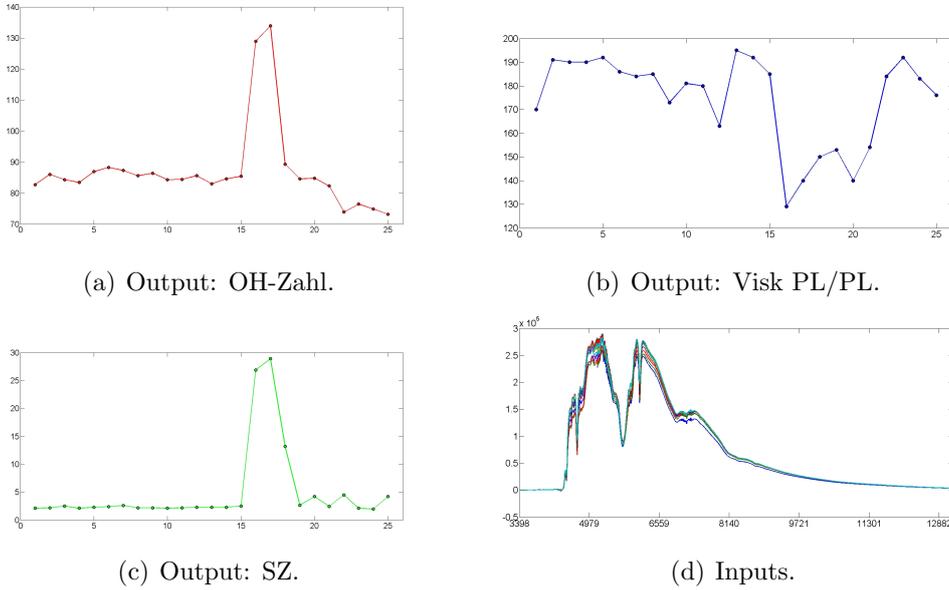


Figure 1: Graphic description of KCCS data set.

We consider three independent targets corresponding, in each of the moments, to

- The *hydroxyl number* (OH-Zahl).
- The *viscosity* (Visk PL/PL), obtained by means of a Parallel Plate Viscometry procedure.
- The *acidity* (SZ).

Figure 1 shows respectively the target values for the 25 multiple instances, in 1(a), 1(b) and 1(c), and the 25 mean values as a representation of the multiple instances, in 1(d).

2.3. Experimental Setup

Firstly we calculate the error using all the variables and we keep it as a threshold for our selection procedure, as well as to be shown in the graphs. The maximum number of generations is 30 and the minimum number of selected individuals is 32. The performance obtained by 1-point crossover is very poor compared with the rest, therefore it is omitted in the results section.

From previous studies [1] over the same data set we have results obtained, by means of cross validation, using several state of the art linear and nonlinear methods such as Multiple Linear Regression (MLR), Principal Components Regression [14], Partial Least Squares Regression [8], Locally Weighted Regression [2], Regression Trees [9], Stepwise Regression [4], Artificial Neural Networks [12], Generalized Linear Models Networks [10], and Flexible Fuzzy Inference Systems [15]. In the graphs we also provide the error obtained using the best of them in each of the outputs for informative purposes.

3. Our Designed GA

As we want to reduce dimensionality we fix a maximum percentage of used variables, i.e. ones, in the individuals of the initial population, which are created randomly. The number of individuals is set to 64. The selection criteria consists on ignoring individuals that perform worse than the "biggest" individual $(1, \dots, 1)$, keeping a minimum number of individuals. We will consider the most popular crossover operator, 2-point crossover, as basic operator to compare with our designed ones. The purpose of designing specific crossover operators is to increase the convergence speed towards good solutions because the evaluation in terms of the fitness function takes a long time. Then the mutation probability was set quite low, 0.5%, because the individuals are supposed to be good solutions. The maximum number of generations has been set to 30.

The chosen model for the fitness function is Partial Least Squares(PLS). PLS finds factors that capture the greatest amount of variance in the predictor variables while they are highly correlated with the predicted variables. Thus PLS attempts to find factors which both capture variance and achieve correlation by means of projecting not only the predictor variables (like PCA), but also the predicted ones, to new spaces so that the relationship between successive pairs of scores is as strong as possible. Considering matrix notation, we would have

$$X = TP + E \quad , \quad y = Tq + f$$

where T is the scores matrix, P and q are the loadings matrix and loadings vector for the inputs X and the output y respectively, and E and f represent the residual errors in both projections. In summary, PLS consists on a search of directions in the factor space that are associated with high variation in the responses but biasing them toward directions that are accurately predicted.

3.1. Forward Selection Crossover

Forward Selection(FS) [11] is a simple data-driven model building approach. In this approach, one adds variables to the model one at a time. At each step, each variable that is not already in the model is tested for inclusion in it. The most significant of these variables, in terms of p-value, is added and its importance is stored. The contribution of the included variable is subtracted, and the next most significant variable is found. At the end all variables are ordered in terms of their significance, being available also the contribution of each of the variables. Note that this multiple use of hypothesis testing means that the real type I error rate for a variable (i.e., the chance of including it in the model given it isn't really necessary), does not equal the critical level we could calculate over the original set. In fact, because of the complexity that arises from the complex nature of the procedure, it is essentially impossible to control error rates and this procedure must be viewed as exploratory, i.e., as a tool for ranking the variables.

Our crossover operator works in a similar way as 2-point crossover but instead of selecting the two points randomly, we select one randomly and the other one in a way so that at least 33% and at most 66% of the total weight of the variables is between both points. The weight is calculated as follows: we determine the $\alpha\%$ most important variables and they all receive maximum weight, i.e. 1. The rest of variables receive a weight in $[0.5, 1)$ by means of linear interpolation. Graphically it is similar to a triangular fuzzy partition centered on the important variables.

3.2. Correlation Crossover

Following the same idea showed before, we determine a first cut point randomly and then we restrict the selection of the second point using as weights the correlations between the variables and the target. That is, we calculate previously a vector containing all the correlations between each variable and the target. After determining one point randomly, we restrict the search of the second one to those parts which leave between both points a percentage of minimum 33% and maximum 66% of the total accumulated weight.

4. Results

Firstly we will see some graphical results target by target and operator by operator. Afterwards we will use the information obtained in the variable selection process, i.e. the reduced data sets using all three crossover operators, in order to check how the performance is when applying FLEXFIS algorithm for each of the samples.

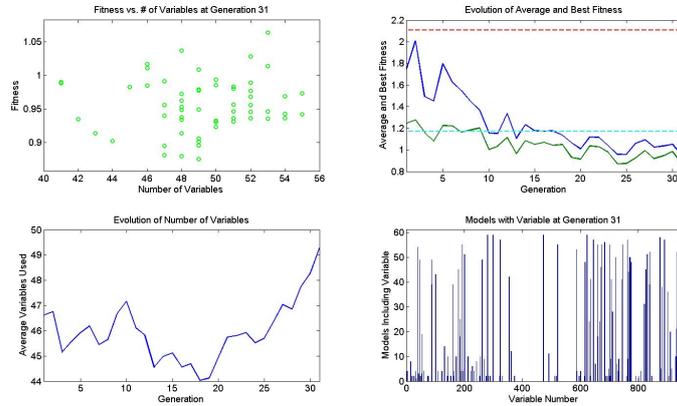


Figure 2: Graphics for 2-point crossover operator on OH.

4.1. GA on OH-Zahl

Figures 2, 3 and 4 present (for 2-point, FS and Corr crossover operators respectively) the values of the fitness function for the different number of variables (top left) and the number of individuals containing each variable (bottom right) at the end of the process, as well as the average number of variables used (bottom left) and the information concerning the errors (top right) for each generation. This last shows the error considering all variables in terms of the fitness function (dotted red line), the best S-o-A algorithm error (dotted cyan), the average error of all individuals (blue line) and the best individual error (green line).

First we should notice that the new operators behave in a more stable way and converge faster than 2-point to good solutions. The number of variables is also lower. When it comes to the designed ones, Corr is more stable, needs less variables and is faster to converge to good solutions than FS.

4.2. GA on Visk PL/PL

Figures 5, 6 and 7 present again (for 2-point, FS and Corr crossover operators respectively) the values of the fitness function for the different number of variables (top left) and the number of individuals containing each variable (bottom right) at the end of the process, as well as the average number of variables used (bottom left) and the information concerning the errors (top right) for each generation.

Now the situation is a bit different in terms of stability, being 2-point and Corr much more stable than FS which otherwise is the more accurate. The number of variables behaves in the same way as for OH.

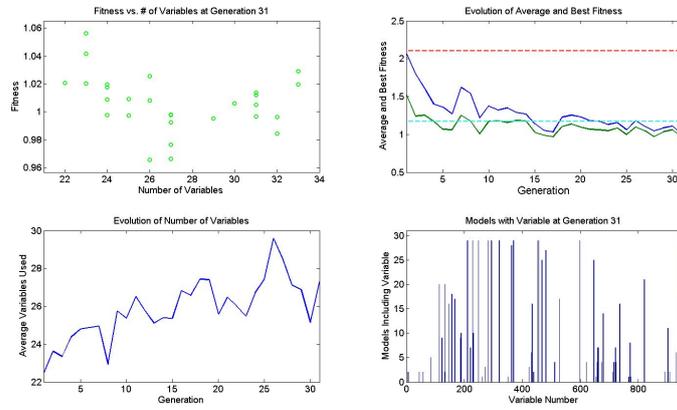


Figure 3: Graphics for Forward Selection crossover operator on OH.

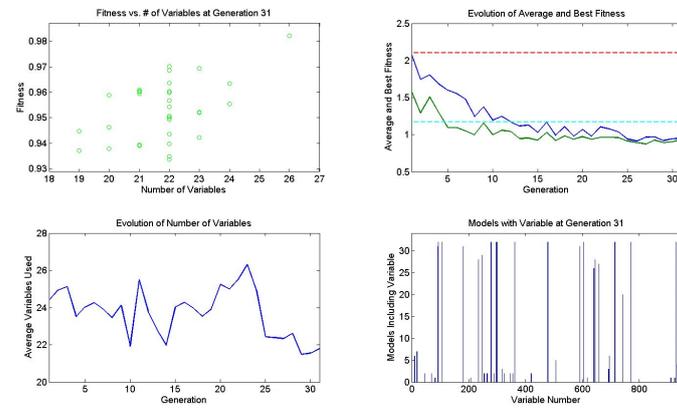


Figure 4: Graphics for Correlation crossover operator on OH.

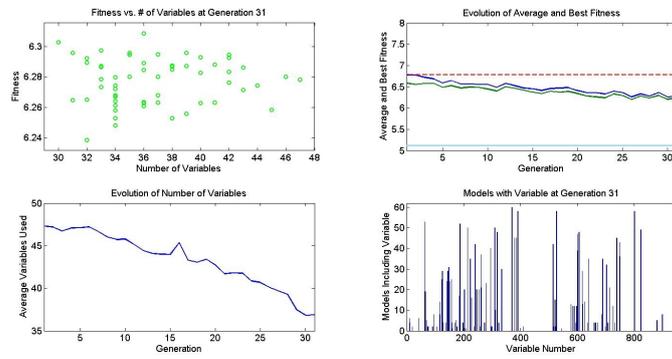


Figure 5: Graphics for 2-point crossover operator on Visk PL/PL.

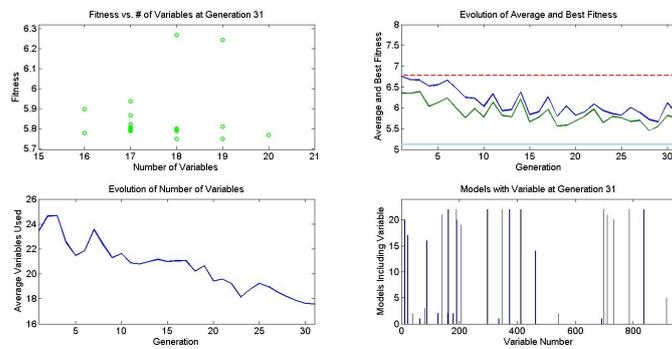


Figure 6: Graphics for Forward Selection crossover operator on Visk PL/PL.

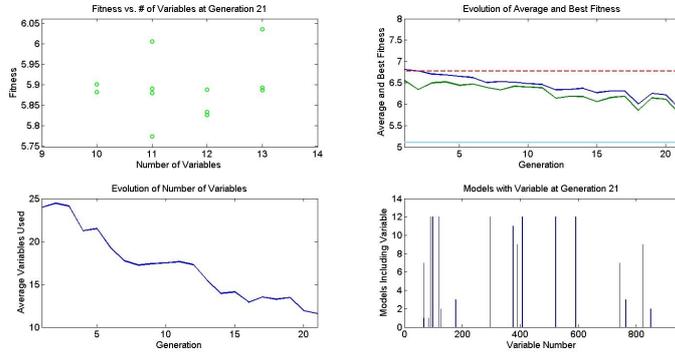


Figure 7: Graphics for Correlation crossover operator on Visk PL/PL.

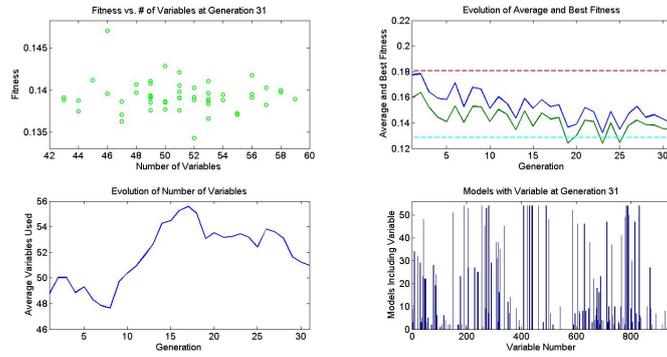


Figure 8: Graphics for 2-point crossover operator on SZ.

4.3. GA on SZ

As before, Figures 8, 9 and 10 present (for 2-point, FS and Corr crossover operators respectively) the values of the fitness function for the different number of variables (top left) and the number of individuals containing each variable (bottom right) at the end of the process, as well as the average number of variables used (bottom left) and the information concerning the errors (top right) for each generation.

The speed for converging is similar for all of them. 2-point obtains the most accurate result but suffers from instability. In general, FS is the best one because it is quite stable and accurate keeping a low number of variables.

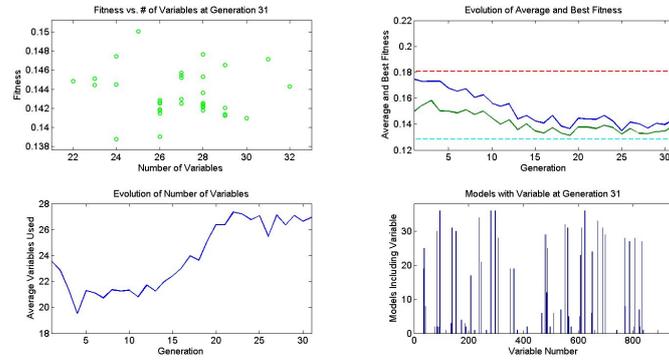


Figure 9: Graphics for Forward Selection crossover operator on SZ.

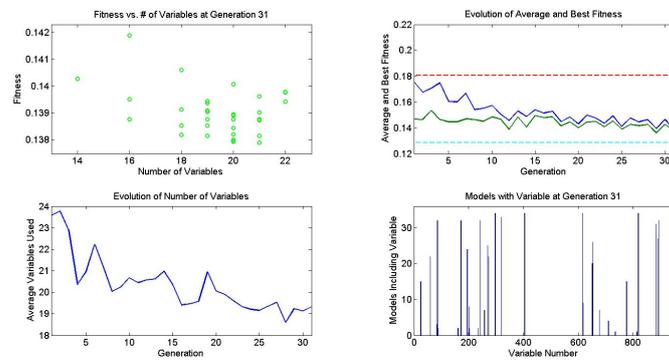


Figure 10: Graphics for Correlation crossover operator on SZ.

| | OH | | | Visk | | | SZ | | |
|------------|-----|--------|--------|------|--------|--------|-----|--------|--------|
| | # | RMSE | SD | # | RMSE | SD | # | RMSE | SD |
| Original | 948 | 1.1715 | 0.5377 | 948 | 5.1028 | 3.0114 | 948 | 0.1285 | 0.0703 |
| 2-p-XOver | 121 | 1.5822 | 0.5997 | 142 | 5.4760 | 2.7895 | 110 | 0.1215 | 0.0602 |
| FS-XOver | 64 | 1.0344 | 0.5083 | 63 | 5.4117 | 2.8842 | 28 | 0.1346 | 0.0491 |
| Corr-XOver | 44 | 0.8194 | 0.3439 | 38 | 5.5260 | 3.2723 | 18 | 0.1441 | 0.0740 |

Table 1: Summary table.

4.4. FLEXFIS Performance

In order to construct a reduced data set with the information provided by the GA, we take the individuals in the pool of solutions after the GA and we keep the variables that are considered in any of the individuals. Table 1 shows the number of variables considered (#), the root mean Square Error (RMSE) and the Standard Deviation (SD) for all three targets and all four data sets, i.e. the original non reduced and the three reduced ones.

We can see that the performance when using the reduced data sets is at least as good as with the original one. For OH it is remarkable the decrease in number of variables, root mean square error and standard deviation wrt the rest. When comparing the results for the reduced data sets only we can notice the importance of the designed operators, the errors are again similar (but in OH for which 2-point behaves worse) with a reduction in the number of variables of between 2 and 8 times wrt 2-point.

5. Conclusion

Problem specific designed crossover operators behave better than standard 1-point and 2-points crossover operators in terms of stability, number of variables and convergence speed. Taking into account that the new ones are quite similar in performance then correlation crossover would be preferable because it is less time consuming. In the future it could be interesting to design more new crossover operators as well as new selection, mutation and initial population generation strategies in order to take advantages of the information provided by the data.

Bibliography

References

- [1] C. Cernuda, E. Lughofer, W. Maerzinger, and Juergen Kasberger. Nir-based quantification of process parameters in polyetheracrylat (pea) production using flexible non-linear fuzzy systems. *Chemometrics and Intelligent Laboratory Systems*, 109(1):22–33, 2011.
- [2] W.S. Cleveland and S.J. Devlin. Locally weighted regression: An approach to regression analysis by local fitting. *Journal of the American Statistical Association*, 84(403):596–610, Sep. 1988.
- [3] J. L. Crosby. Computers in the study of evolution. *Sci. Prog. Oxf.*, 55:279–292, 1967.
- [4] N.R. Draper and H. Smith. *Applied regression analysis*. Wiley Interscience, Hoboken, NJ, 1998.
- [5] D. B. Fogel. *Evolutionary computation: The fossil record*. IEEE Press, 1998.
- [6] A. S. Fraser. Simulation of genetic systems by automatic digital computers: I. introduction. *Australian J. Biol. Sci.*, 10:484–491, 1957.
- [7] D. Goldberg. *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley, 1989.
- [8] M. Haenlein and A.M. Kaplan. A beginner’s guide to partial least squares (PLS) analysis. *Understanding Statistics*, 3(4):283–297, 2004.
- [9] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference and Prediction - Second Edition*. Springer, New York Berlin Heidelberg, 2009.
- [10] T. Hastie, R. Tibshirani, and J. Friedman. Regularized paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1), 2010.
- [11] Friedman J. Hastie T., Tibshirani R. *The Elements of Statistical Learning*. Springer, New York, 2001.
- [12] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 1999.

- [13] J. H. Holland. *Adaptation in natural and artificial systems*. *Univ. Michigan Press*, 1975.
- [14] I.T. Jolliffe. *Principal Component Analysis*. Springer Verlag, Berlin Heidelberg New York, 2002.
- [15] E. Lughofer. FLEXFIS: A robust incremental learning approach for evolving TS fuzzy models. *IEEE Trans. on Fuzzy Systems*, 16(6):1393–1410, 2008.
- [16] J. Reed, R. Toombs, and N. A. Barricelli. Simulation of biological evolution and machine learning: I. selection of self-reproducing numeric patterns by data processing machines, effects of hereditary control, mutation type and crossing. *J. Theoret. Biol.*, 17:319–342, 1967.

Improving the Parsimony of Regression Models for an Enhanced Genetic Programming Process

Alexandru-Ciprian Zăvoianu¹, Gabriel Kronberger², Michael Kommenda²,
Daniela Zaharie¹, Michael Affenzeller²

¹ Department of Computer Science, West University of Timișoara, Romania

² Heuristic and Evolutionary Algorithms Laboratory (HEAL), Upper Austrian
University of Applied Sciences, Austria

Abstract. This research is focused on reducing the average size of the solutions generated by an enhanced GP process without affecting the high predictive accuracy the method exhibits when being applied on a complex, industry proposed, regression problem. As such, the effects the GP enhancements have on bloat have been studied and, finally, a bloat control system based on dynamic depth limiting (DDL) and iterated tournament pruning (ITP) was designed. The resulting bloat control system is able to improve by $\simeq 40\%$ the average GP solution parsimony without impacting average solution accuracy.

Keywords: genetic programming, symbolic regression, solution parsimony, bloat control

1 Introduction

For the concrete, industry proposed, system identification problem that has been considered for this research, an enhanced GP algorithm (Ehd-GP) developed by the HEAL team is able to produce high quality regression models that were of comparable, if not even better, accuracy than the regression models obtained using other, more well established, non-linear, data-mining techniques like support vector machines (SVMs) and artificial neural networks (ANNs) [6] [11]. Further evidence of the high quality of the solutions generated by Ehd-GP is summarized in [9].

The main advantage of using GP for the given regression problem task lies in the the ability of the this method to produce white-box, human interpretable, models than can be easily used by human domain experts to gain new insight into phenomena associated with the given industrial process. However, the degree of interpretability a GP generated regression model exhibits is proportional to its parsimony as trying to analyze large and/or highly complex models can be a quite tedious task.

A general threat to GP model interpretability comes in the form of a well studied phenomenon known as *bloat*: a generation-wise rapid growth in the size of evolved programs without any corresponding benefits in terms of accuracy. This phenomenon is very well known to GP practitioners and literature proposes

a large set of studies regarding bloat. Controlling and combating bloat is an open and fairly complicated task as Silva and Costa [8] argue that the phenomenon is very likely to be a natural consequence of combining an evolutive (fitness driven) search with a variable-length solution representation (i.e. two of the main characteristics of the GP theory).

With the above mentioned evidence of the ability of Ehd-GP to produce high accuracy regression models, the purpose of this research is to analyze the impact the modifications in Ehd-GP have on bloat and, consequently, try to control the bloating phenomenon in this GP process such as to reduce the average size of the resulting solution models without affecting the high average quality of these models.

2 The Ehd-GP Process

The Offspring Selection Strategy (OS) The most important modification the Ehd-GP proposes over the standard GP process described by Koza [5] is the incorporation of the offspring selection strategy [1]. The idea behind this elitist selection method is that an offspring is accepted as a member of the population of the next generation if and only if it outperforms the fitness of its own parents. As with conventional GP, offspring are generated by parent selection, crossover and mutation. The difference consists in the introduction of a second (offspring) selection stage. A variable called success ratio indicates the ratio of the next generation members that must outperform their respective parents. As long as this ratio is not fulfilled, offspring are created and the successful ones are inserted into the next generation, while those that do not outperform their parents are stored in a rejection pool. After enough successful next generation offspring have been created, the rest of the members of the generation are randomly chosen from the rejection pool.

The Linear Scaled Error Measure The second important enhancement that Ehd-GP proposes is the use of a linear scaled error measure (as described by Keijzer in [4]) in the fitness evaluation function. The advantage of using a scaled mean squared error (MSE), as opposed to the traditional approach, lies in “the change of the view” that the selection operator has on the worth of an individual expression. As this error measure rescales the expression on the optimal slope and intercept, selection will favor expressions that are close in shape with the target, instead of demanding first that the scale is correct.

3 Test Setup

The industry proposed main modeling scenario used in our tests contains 44 dependent variables and has been split into a training set, a validation set and a test set. For a detailed description of the benefits of using this data partitioning strategy in GP based modeling please see [9].

Methodology Because GP remains at its core a stochastic method, we have performed 100 GP runs of a given configuration in order to determine the general behavior of that configuration (in terms of solution size and accuracy). We shall refer to the data set that contains all the 100 solutions obtained for a specific GP configuration as the *the full data set* of that configuration. Since solution accuracy is the first performance GP criteria, we are especially interested in analyzing and improving the best solutions that can be generated with a given GP configuration. As a result, for each GP configuration, we have also constructed a *top accuracy subset* that only contains the 25 most accurate solutions in the full data set. Ties were broken in favor of the more parsimonious model.

Throughout this paper, our comparisons are based on basic central tendency indicators (the average - μ , the standard deviation - σ and the median - $\mu_{1/2}$) for both the full solution data sets and the top accuracy subsets.

When comparing among GP configuration results based on the full data sets, we also make use of *significance testing* to confirm our empirically based hypotheses. The significance test we use is the *Mann-Whitney-Wilcoxon test* (also know as the Mann-Whitney U test). The used significance level is $\alpha = 0.05$ in the case of one-tailed tests. The choice for this particular non-parametric test was made because we do not wish to presume that our solution data set is normally distributed, either according to size or accuracy.

In order to provide a simple but accurate and suggestive overview of the performance of the GP solutions in the full and top accuracy data sets, for each GP configuration, we construct *comparative kernel density estimation* plots of the the solution size and of the solution MSE.

GP Configurations The Ehd-GP was configured to enforce a strict OS thus forcing every member of the next generation to outperform both its parents. When generating a new offspring, one parent was chosen using proportional selection and the other was chosen using random selection.

For our tests we have also used a Koza style standard GP process (Std-GP) in which both parents were selected according to proportional selection.

Both GP processes used only point mutation and in both cases constants were initialized and modified uniformly (between -20 and 20), whilst variables were initialized and modified according to a normal distribution $N(0,1)$. The other, more common, GP algorithm parameters that were used by both the Std-GP and the Ehd-GP processes are: an arithmetic function library (+, -, *, %, *power*, *sqrt*), a population size of 1000 individuals, a mutation rate of 15%, a max tree height of 15 and a maximum number of evolved generations of 1000.

The initial population initialization method was PTC2 [7] for both GP processes and the stopping criterion stated that a run should be terminated if no validation wise improvement was found in the last 100 generations.

The max tree height limit value was chosen empirically after performing systematic tests on the modeling scenarios considered in this research and observing that the very good solutions have a depth smaller than 10.

4 Ehd-GP performance analysis

The performances of Std-GP and of Ehd-GP with regards to average solution accuracy and average solution size are summarized in Table 1. The corresponding comparative kernel density plots are presented in Fig. 1.

Our first empirical observation that the Ehd-GP is far superior in terms of average solution accuracy is statistically significant as the *Mann-Whitney-Wilcoxon test* yielded a one-tailed *p-value* smaller than 0.001.

The second empirical observation that the Ehd-GP tends to produce solutions of lower parsimony than the Std-GP is also statistically significant, being confirmed as the one-tailed *Mann-Whitney-Wilcoxon test* produced a one-tailed *p-value* of 0.005.

5 A Bloat Control System for Ehd-GP

The main challenge of fitting an efficient bloat control mechanism in Ehd-GP came from trying to integrate this such a modification with the existing OS strategy.

For instance, anti-bloat selection methods [10][3] proved fairly difficult to combine with the offspring selection enhancement because their most effective implementations are based on dynamical adjustments of their control parameters. When using OS with a linear scaled error measure, the GP algorithm converges quite fast (in $> 90\%$ of the runs the solution was found before the 22nd generation) and this means that there are far fewer opportunities for parameter adjustment.

In the case of anti-bloat genetic operators [2], our educated guess is that for Ehd-GP, the combination between size and/or depth limitations imposed to genetic operators (i.e. restricting the number of offspring than can be generated) and OS has a high chance of leading to premature convergence (because of a rapid drop in genetic diversity).

Dynamic Depth Limits - DDL In [8], Silva and Costa present a simple yet effective solution for overcoming most of the shortcomings of static depth limitation. Their idea is to dynamically adjust the value of the depth limit during the run. Compared to the original method, we have made a series of modifications in order to integrate DDL into the Ehd-GP process. In our implementation of the concept (Algorithm 1), the dynamic limit is initially set to a low value (*InitialDepthLimit*), that is usually 20-30% higher than that of the maximum depth in the initial population. An offspring is automatically accepted in the next generation if it satisfies the accuracy constraint and at the same time does not infringe the depth limit. If an offspring infringes the depth limit but is the best individual found so far, then it is accepted in the next generation if the increase in size is matched by the increase in accuracy. In the latter case, the limit is raised to match the depth of the new best-of-the-run individual. If during

Algorithm 1 The dynamic depth limiting module (DDL module)

```

1: AcceptStatus = true
2: if OffDepth ≤ DepthLimit then
3:   if (BestMSE / OffspringMSE - 1) ≥ (DLimit - OffspringDepth) * Clower then
4:     if OffspringDepth > InitialDepthLimit then
5:       DLimit = OffspringDepth
6:     else
7:       DLimit = InitialDepthLimit
8:     end if
9:   end if
10: else
11:   if (BestMSE / OffMSE - 1) ≥ (OffspringDepth - DLimit) * Craise then
12:     DLimit = OffspringDepth
13:   else
14:     AcceptStatus = false
15:   end if
16: end if
17: return AcceptStatus

```

the run the best found individual at a given time has a depth that is significantly lower than the current depth limit, the limit will be lowered.

The condition on Line 11 states that the DDL should be raised if each extra depth level is matched by an increase in training accuracy of at least $C_{raise}\%$. Analogously, the condition on Line 3 states that the DDL should be lowered if each decreased depth level is matched by an increase in training accuracy of at least $C_{lower}\%$. As an empirical rule, tests have shown that the relation $C_{lower} = 2 * C_{raise}$ enables the DDL to have a stable behaviour throughout the run for all the considered test scenarios. Furthermore, after testing on three other modeling scenarios, we discovered that $C_{raise} = 0.015$ is also a stable setting.

Iterated Tournament Pruning - ITP Largely inspired by [9], and taking into consideration all the particularities of the GP process we are trying to enhance, we also decided to implement and test a bloat control strategy based on syntax-tree pruning. Our ITP strategy is described in Algorithm 2 and is based on a series of consecutive pruning tournaments. In each tournament we create several pruning candidates of the syntax-tree model we wish to prune in that tournament (the pruning base model). A pruning candidate for a model is created through a very simple process (Line 6) that randomly selects a subtree from the given model and replaces it with the mean value obtained by evaluating that subtree over all the records in the training set. The size of the excised subtree is limited to at most $MaxPruning\%$ with regards to the pruning base model. At the end of each tournament the pruning candidate with the highest accuracy (minimum MSE) will be selected as the next pruning base.

Initial tests performed with various settings for the configuration parameters ITP supports indicate that: the *MaxPruning* and *IterationsCount* parameters

should be set such as to generally avoid the possibility of reducing a model to a single node; ITP displays the best results when applied only to the individuals that are not among the best 10-15% nor among the worst 20-40% according to accuracy; an increased mutation rate of 25-50% would improve solution accuracy;

Algorithm 2 The iterated tournament pruning module (ITP module)

```

1: BestMSE =  $\infty$ 
2: PrunSolution =  $\emptyset$ 
3: PrunBase = OriginalModel
4: for  $i = 0$  to IterationsCount do
5:   for  $j = 0$  to TournamentSize do
6:     PrunCandidate = StochasticPrune(PrunBase, MaxPruning)
7:     PrunMSE = ComputeMSE(PrunCandidate)
8:     if PrunMSE < BestMSE then
9:       PrunSolution = PrunCandidate
10:      BestPrunMSE = PrunMSE
11:     end if
12:   end for
13:   PrunBase = PrunSolution
14: end for
15: return PrunSolution

```

The Resulting Bloat Control System - BCS From the previous descriptions, one can observe that DDL and ITP have two complementary ways of fighting bloat. Whilst the former tries to prevent the creation and propagation of bloat, the latter directly attempts to remove superfluous code from the population members. As such, trying to combine both methods in a single integrated solution aimed at combating bloating, seemed a very natural approach.

In our initial tests of the combined bloat control system, we decided to use for the two bloat control methods the same parameter settings from the stand-alone configurations. To our surprise, the approach proved quite successful.

6 Conclusions

Individually both the DDL and ITP modules managed to reduce the average Ehd-GP solution size (see Table 1). Both improvements were statistically significant with *Mann-Whitney-Wilcoxon test p-values* smaller than 0.0001.

The combination of the two bloat control methods proved to reduce even more the average Ehd-GP solution size (see Table 1). The average decrease of solution size determined by BCS was statistically significant both with regards to individual DDL (*p-value* = 0.0183) and to individual ITP (*p-value* < 0.0001).

While empirical observations may suggest that BCS also slightly increased the average solution quality of the Ehd-GP, there was no solid statistic proof

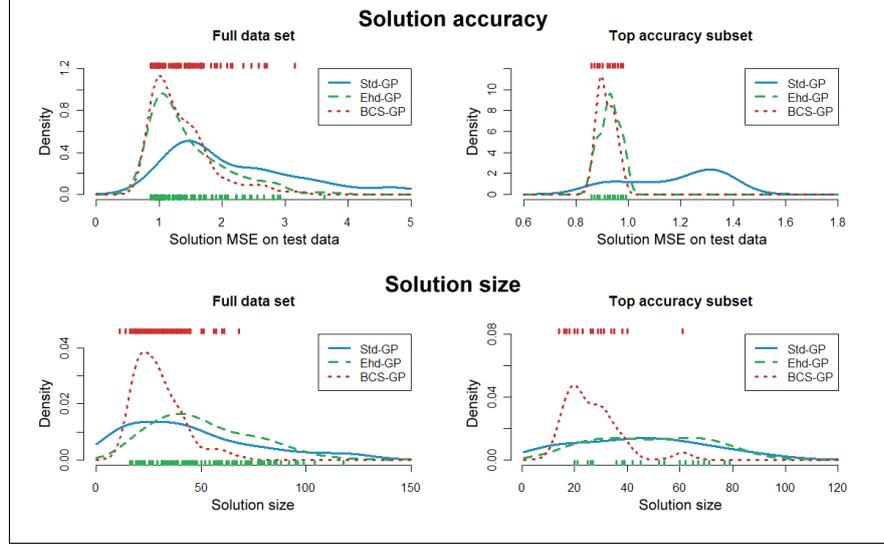


Fig. 1. Kernel density estimation plots for Std-GP and Ehd-GP

Table 1. Accuracy and size information regarding Std-GP and of Ehd-GP solutions

| | Full data set | | | Acc. subset | | |
|-------------|---------------|----------|-------------|-------------|----------|-------------|
| | μ | σ | $\mu_{1/2}$ | μ | σ | $\mu_{1/2}$ |
| Std-GP MSE | 2.178 | 1.046 | 1.821 | 1.177 | 0.177 | 1.232 |
| Ehd-GP MSE | 1.448 | 0.584 | 1.280 | 0.929 | 0.037 | 0.930 |
| DDL-GP MSE | 1.514 | 0.643 | 1.345 | 0.919 | 0.049 | 0.940 |
| ITP-GP MSE | 1.228 | 0.925 | 1.065 | 0.962 | 0.035 | 0.970 |
| BCS-GP MSE | 1.339 | 0.470 | 1.210 | 0.915 | 0.032 | 0.900 |
| Std-GP size | 45.44 | 33.50 | 39.00 | 44.84 | 24.55 | 43.00 |
| Ehd-GP size | 52.60 | 28.11 | 47.00 | 53.36 | 29.94 | 52.00 |
| DDL-GP size | 32.92 | 16.30 | 30.50 | 34.16 | 14.32 | 34.00 |
| ITP-GP size | 39.34 | 18.65 | 35.00 | 42.16 | 18.43 | 38.00 |
| BCS-GP size | 29.17 | 11.18 | 28.00 | 26.36 | 10.21 | 23.00 |

that the average solution accuracy of the BCS augmented Ehd-GP process (BCS-GP) was any different from that of original Ehd-GP. The BCS-GP process was further tested on two more symbolic regression problems in [11] and the results confirmed the findings reported in this paper. The corresponding comparative kernel density plots of Ehd-GP and BCS-GP are presented in Fig. 1.

In spite of the very good results obtained by the final bloat control system, we consider that, at the current stage of development, its main function is that of a proof of concept with regards to the successful combination of two bloat combating methods that are based on different but complementary paradigms.

Acknowledgments

The work described in this paper was done within the Josef Ressel Centre for Heuristic Optimization Heureka! and sponsored by the Austrian Research Promotion Agency (FFG). For more information about Heureka! please visit <http://heureka.heuristiclab.com>.

The stay of Mr. Alexandru-Ciprian Zăvoianu at the Heuristic and Evolutionary Algorithms Laboratory has been supported by the EU through the SPRERS project (FP7 reference number 246839).

References

1. Affenzeller, M., Wagner, S.: Offspring selection: A new self-adaptive selection scheme for genetic algorithms. In: Ribeiro, B., Albrecht, R.F., Dobnikar, A., Pearson, D.W., Steele, N.C. (eds.) *Adaptive and Natural Computing Algorithms*. pp. 218–221. Springer (2005)
2. Crawford-Marks, R., Spector, L.: Size control via size fair genetic operators in the PushGP genetic programming system. In: *GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference*. pp. 733–739. Morgan Kaufmann Publishers, New York (9-13 Jul 2002)
3. Ekart, A., Nemeth, S.Z.: Selection based on the pareto nondomination criterion for controlling code growth in genetic programming. *Genetic Programming and Evolvable Machines* 2(1), 61–73 (Mar 2001)
4. Keijzer, M.: Scaled symbolic regression. *Genetic Programming and Evolvable Machines* 5(3), 259–269 (Sep 2004)
5. Koza, J.R.: *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA (1992)
6. Kronberger, G., C.Feilmayr, Kommenda, M., Winkler, S., Affenzeller, M., Burgler, T.: System identification of blast furnace processes with genetic programming. In: *Logistics and Industrial Informatics - LINDI*. pp. 1–6. IEEE Press (2009)
7. Luke, S.: Two fast tree-creation algorithms for genetic programming. *IEEE Transactions on Evolutionary Computation* 4(3), 274–283 (Sep 2000)
8. Silva, S., Costa, E.: Dynamic limits for bloat control in genetic programming and a review of past and current bloat theories. *Genetic Programming and Evolvable Machines* 10(2), 141–179 (2009)
9. Winkler, S.M.: *Evolutionary System Identification*. Ph.D. thesis, Johannes-Kepler-Universität, Linz, Austria (2008)
10. Zhang, B.T., Mühlenbein, H.: Balancing accuracy and parsimony in genetic programming. *Evolutionary Computation* 3(1), 17–38 (1995)
11. Zăvoianu, A.C.: *Towards solution parsimony in an enhanced genetic programming process*. Master’s thesis, International School Informatics: Engineering & Management, ISI-Hagenberg, Johannes Kepler University, Linz (2010)

Local Structural Similarities Based on Distance Transforms for Image Comparison

November 22, 2011

Jean-Luc Bouchot - Frederic Morain-Nicolier
email: jean-luc.bouchot@jku.at

Abstract — In this talk we introduce the use of different distance transforms for the evaluation of local structural (dis-)similarities. While structural similarities as the SSIM index have shown growing interest in the last decade, we investigate the use of different distance transforms to create index of dissimilarity between images.

We will first develop the main work dating back to the early 90s with a distance transform basically used for assessing the quality of an edge detector on binary images.

Then we introduce some extension of the binary distance transform to gray-level images and define a metric between distance transforms which we use for image comparison.

The feasibility of the proposed approaches is studied under some natural scenes examples.

Key words — *Distance Transform, Structural Similarity, Local Analysis, p-norm approximation*



1 Introduction

Distance measure between images is of theoretical as well as practical importance. The always increasing amount of images available on the web as well as the development of online and offline image retrieval tasks with many different images makes the design of a good similarity measure hard and interesting. These similarities are then used, for instance, for assessing the image quality, for segmenting an image, for compression algorithm evaluation. In this talk we survey the use of different distance transform based measure and introduce a new one. Our work is articulated as follows. We start by recalling the basics principle of binary image distance transforms and their use for comparing binary images. Section 3 extends the distance transform to the gray level case, and the image comparison metrics are introduced in Section 4. Finally some first experiments on natural scenes compared with some state of the art structural similarity measures are given in the final part of the talk.

2 Error Metric for Binary Images

This section aims at introducing a previous work from Baddeley [1]. He introduced a metric based on the distance transform and Hausdorff distance. It is only used for binary images in the context of image comparison. We first review how the distance transform is computed and then the implementations for its linear computation/approximation. The last section introduces the metric used for image comparison which is based on a L_p norm approximation of the Hausdorff distance.

2.1 Distance Transform

The basic idea for distance transforms in binary image processing is to convert any binary pattern into a distance map which contains the distance to the closest foreground pixel. Therefore we need to specify a distance d for measuring closeness in the pixel grid and we can define the distance map as

Definition 1 (Binary distance map). *Given a binary signal f defined on \mathbb{R}^N , which can actually be represented as a set $A = \{x \in \mathbb{R}^N : f(x) = 1\}$, the distance transform is defined as*

$$\forall x \in \mathbb{R}^N, DT_d[f](x) = d(x, A) = \min_{y \in A} (d(x, y))$$

This definition yields a structural description of the image as contours are obtained easily by taking the following sets: $A_k = \{x \in \mathbb{R}^N : DT_d[f](x) = k\}$. This example defines all the points of a binary image lying at a distance of k to the closest foreground element.

2.2 Algorithms

Interesting with this representation is the cheap way to compute it. Some methods have been developed to compute some approximations of the Euclidean Distance Transform which are no more than 2% wrong. We refer the reader to [9, 3] for some theoretical concepts and proofs of this results as well as the mask used. It is important to notice that thanks to this approximation,

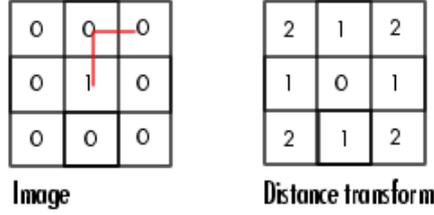


Figure 1: Sample binary distance transform with L_1 distance measure.

the euclidean distance transform of an image can be approximated simply with passing a mask forward and backward on an image.

Implementation of such forward and backward mask passing for higher dimensions have been developed too. See for instance [4]. This formulation might help extending all the proposed approaches to higher dimensional signals (for instance imaging in biomedicine).

2.3 Image Metric

Baddeley based his work on the use of the Hausdorff metric, which is actually a distance between sets.

Definition 2 (Hausdorff distance). *Given two sets $A, B \subset \mathbb{R}^N$, the Hausdorff distance is defined as*

$$H(X, Y) = \max(\sup_{x \in A} (d(x, B)), \sup_{y \in B} (d(y, A)))$$

Baddeley introduced the use of the Hausdorff distance for binary image comparison. Indeed, as stated in definition 1, any binary image can be written as a set. However the Hausdorff distance is not usable in its original form and an equivalent formulation has been proposed:

$$H(X, Y) = \sup_{x \in X} |d(x, A) - d(x, B)| \quad (1)$$

Now it appears that the Hausdorff distance can be implemented with distance transforms algorithms and is therefore tractable for image processing applications. One last point remains critical regarding noise sensitivity. Because of the max or sup calculation the presence of strong outliers will introduce a bias in the measurement of dissimilarity. The idea given by Baddeley to overcome this problem is to give an averaged approximation of the Hausdorff distance based on p -norm which yields his dissimilarity coefficient, defined, for a given integer q as

$$\Delta_b^q(f, g) = \left[\frac{1}{N_X} \sum_{x \in X} |d(x, A) - d(x, B)|^q \right]^{1/q} \quad (2)$$

where A and B are the two sets respectively associated to the signals f and g and N_X denotes the number of points in the pixel grid.

3 Gray-Level Distance Transforms

As detailed in the previous section, a metric for binary image comparison was introduced based on a distance transform and p -norm approximation. Since then, different methods have been proposed as an extension to the gray level case. Some work try to extend the Δ_b metric itself (see Section 3.1) while other approaches try to extend the distance transform (see Section 3.2).

Other methods such as Weighted Distance Transform on Curved Spaces or the Grey Weighted Distance Transform won't be detailed here (the curious reader can have a first look at [7]). Neither will be the work of Coquin and Bolon [5], or the one of Felzenszwalb and Huttenlocher [6].

3.1 Wilson

The approach introduced by Wilson *et al.* [12] aims at extending the previous Δ_b metric to a metric capable of handling gray-level images. The idea is to embed an N -Dimensional signal into a $N+1$ dimensional space and to compute distances from each point of this space to the subgraph of the signal.

Definition 3 (Subgraph of a signal). *Given a signal f defined on $X \subset \mathbb{R}^N$ whose values are in a given set Y (i.e. $f : X \rightarrow Y$), its subgraph Γ_f is defined as*

$$\Gamma_f = \{(x, y) \in X \times Y : y \leq f(x)\}$$

Note that N will typically have value 1 (for proper signals) or 2 (for images). Y is the set of possible gray values of the signal; for instance, for an 8-bit image we have $Y = \{0, \dots, 255\}$.

From now on, if needed we will denote by *spatial coordinates* any point in X and by *intensity* or *illumination* values in Y .

Now, let D be the containing space, i.e. $D = X \times Y$, we compute, for each point $p = (x, y) \in D$ the distance to the subgraph of f :

$$d(p, \Gamma_f) = \inf_{p' \in \Gamma_f} d(p, p') \quad (3)$$

Remark that expression 3 is general and one needs to specify the distance used inside the inf. Wilson *et al.* have proposed a distance that separately takes into account spatial coordinates and intensity values.

Definition 4 (Wilson's distance). *We can define a distance on $X \times Y$ as*

$$\forall p = (x, y), p' = (x', y') \in D, d_W(p, p') = \max(d(x, x'), |y - y'|)$$

where $d(x, x')$ defines, for instance, the euclidean distance between the spatial coordinates.

We can now introduce the notion of upper level set, which is a usefull concept for the derivation of a metric which makes use of distance transform as for the binary case.

Definition 5 (Upper-level set). *Given a signal f , we define an upper-level set at y as*

$$X_y(f) = \{x \in X : f(x) \geq y\}$$

Finally plugging definitions 4 and 5 into equation 3 yields Wilson's gray level distance transform:

$$\forall p = (x, y) \in D, DT_W[f](p) = d(p, \Gamma_f) = \inf_{z \in Y} \{\max(d(x, X_z(f)), |y - z|)\} \quad (4)$$

Finally only a finite number of binary distance transforms have to be computed to get Wilson's gray level distance transform.

However, we propose another way to write this down, as it is rather unclear how to understand, in a physical sense, the distance used here. Indeed, if we consider a 1-dimensional signal, then the set X is a set of given time stamps while Y represents quantitative values (price of an action for instance) and one should be careful when taking a max on variables with different signification. Therefore we propose to add a normalization parameter which should be there to make sure both spatial and intensity variables are comparable. This yield the following normalized distance:

$$d_W^\rho(p, p') = \max(d(x, x'), \rho|y - y'|) \quad (5)$$

Embedding a signal into a higher dimensional space has another drawback. It makes the visualization of the distances hard. Indeed, considering a 2-Dimensional signal (an image) the distance transform using the approach of Wilson *et al.* yields a 3-Dimensional map which is no longer easily displayed.

A last idea suggested is to cut off values higher than a threshold c in the definition of the distance transform.

Definition 6 (Thresholded distance). *Let c be a positive real number. The following expression can be used as a distance for the distance transform:*

$$d_W^{\rho *}(p, p') = \min\{d_W^\rho(p, p'), c\}$$

This formulation allows to control the highest value achievable in the distance transform, as we will see later, and to eventually accelerate the computation. Indeed, by considering a thresholding of the distance, we can directly get rid of all the distance transform appear at an intensity distance of more than the thresholding (more details are given in [12]).

As we see on Fig. 2, both c and ρ parameters strongly impact the results of Wilson's Distance Transform. If we consider the extrem case $\rho = 0$, we get that $d_W^{\rho *}(p, p') = \min\{d_W^0(p, p'), c\} = d(x, x')$. And because we are taking the infimum of this value to all possible cut, this would yield a value of 0 everywhere but at the max of the signal. This is actually what we see when we look at a given row of the grid from right to left: the images are getting darker and darker until only a single white bar located at the signal's maximum is visible.

On the other hand, taking a too large value for ρ will contribute to blow away all the spatial coordinates and help the dissimilarity values to surpass the saturation threshold c . This is what we see upper right triangle of the grid, where many values obtained are completely bright white and therefore contains almost no more structural information (apart from the fact that the closest cut more than c away from the given point). Of course, the larger the c the larger ρ can be. Of course in this case "too large value" of ρ strongly depends on the chosen saturation threshold.

Finally, c allows to choose which kind of structure we want to detect between rather local structure (small c) or more global features (larger c), as we can see, for instance in the third column of the grid.

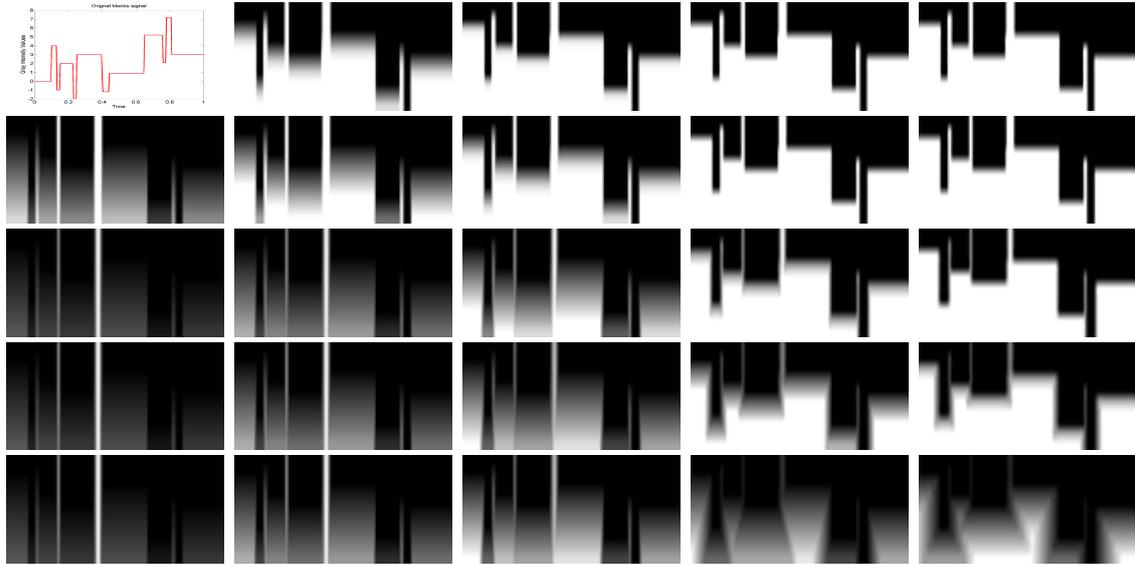


Figure 2: Influence of the different saturation and normalization parameters of Wilson's Distance Transform. Each row fixes a c (1,2,5,10,100, from top to bottom) while each column fixes a ρ parameter (0.5,1,2,8,16 from left to right). The upper left corner shows the signal used for the transformation. The reference signal is actually of size 256 and ranges from about -2 to 8. Some cuts were made every intensity unit.

3.2 Molchanov

Another extension to higher dimensional space have been proposed by Molchanov [8]. In his work the author introduces a distance transform which is also based on upper-level sets and sums them together with some predefined weight. Molchanov does not consider the illumination coordinates anymore (except intrinsically for the definition of the upper level sets). This simplifies the choice for a normalization factor, compared to Wilson's approach.

Definition 7 (Molchanov's Distance Transform). *Given a real-valued signal f defined on a domain $X \subset \mathbb{R}^N$, we can define a real-valued distance transform as follows:*

$$DT_M[f](x) = \sum_{y \in Y} \mu_y d(x, X_y(f))$$

where d denotes the euclidean distance on the spatial coordinates and $\{\mu_y\}_{y \in Y}$ denotes some weighting factors.

The interesting point with this definition is that it does not create any artifact for the visualization by adding a dimension to the signal. Moreover it has been well developed in a mathematical sense and allows one to invert this procedure (at least for discrete finite signals; which is the case for images); it means that one can compute a distance transform map from an image or get an image from a distance transform map.

Figs. 3(b) to (d) show the impact of discretizing the intensity space over different values. It appears that the structure of the image with higher number of intensity levels is very similar to the one with low number of intensity levels.

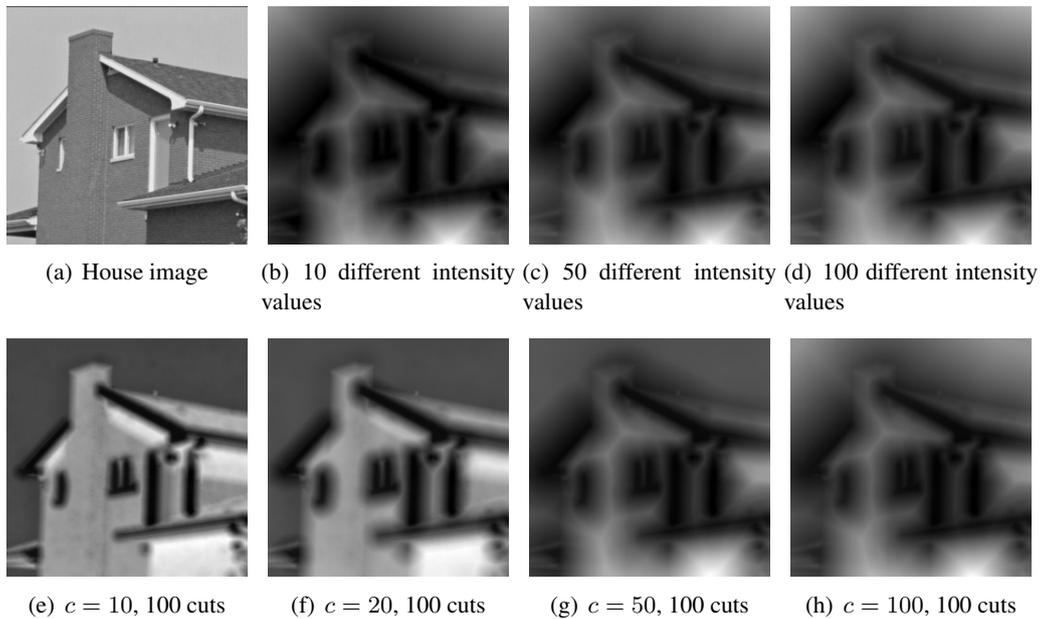


Figure 3: Molchanov distance transform on the House image. First row show the results with varying number of discretization steps of the intensities while second row show varying value of thresholding parameter.

As before, we can introduced a thresholded variant of the given distance: every distances higher then a given threshold c are fixed to c . This allows a better control on the range of values achievable by the distance transform. It also allows to consider dissimilarity within a circle of a given radius around each pixel. Figs. 3(e) to 3(h) show how different kind of details appear when using different thresholding values. It seems that the lower the thresholding value, the finer the details appearing, while larger values of threshold yield coarser details.

Figs. 4 shows another example of different threshold on a more complex image. While the House image has low textural information (only the brick wall), the Lena image shows interesting textures, for instance in the hair or in the hat. On this image too it appears that the choice of c affects the kind or size of details considered. On the Lena example, it appears that with a lower c more strength is given to the high frequency details, while high values of c emphasizes on the global shape of the face.

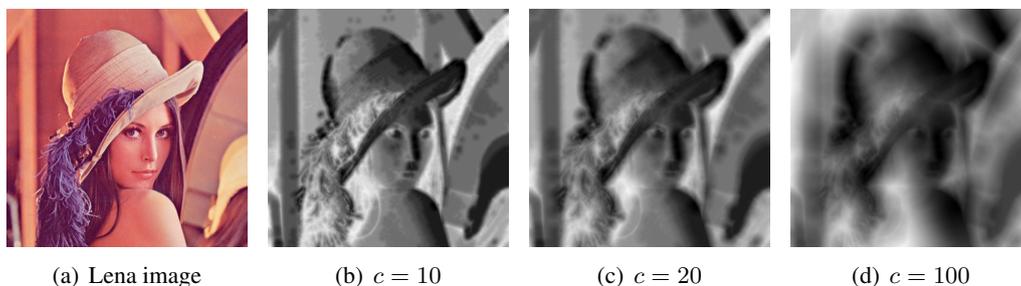


Figure 4: Molchanov distance transform with different thresholdings of the distances on the Lena image.

4 Gray-Level Distance Metric

4.1 Global dissimilarity metrics

Baddeley combined the distance transform of two images to compute a metric for comparison as developed in sec. 2.3. This idea has also been applied in the work of Wilson *et al.* for the comparison of gray level images. This yields the following coefficient, as an approximation of a max with q -norms:

Definition 8 (Wilson's metric for gray-level images). *Given two signals f and g , Wilson's dissimilarity coefficient is given as*

$$\Delta_W^q(f, g) = \left[\frac{1}{N_D} \sum_{p \in D} |DT_W^\rho[f](p) - DT_W^\rho[g](p)|^q \right]^{1/q}$$

where N_D denotes the number of points in the considered $N+1$ dimensional pixel raster D .

To our knowledge, no other such measures have been built for image comparison based on Molchanov's distance transform 7. Therefore we propose a similar coefficient:

Definition 9 (Molchanov's metric for gray-level images). *Given two signals f and g , Molchanov's dissimilarity coefficient can be given as*

$$\Delta_M^q(f, g) = \left[\frac{1}{N_X} \sum_{x \in X} |DT_M[f](x) - DT_M[g](x)|^q \right]^{1/q}$$

where N_X denotes the number of points in the considered spatial pixel raster.

Both metrics described above are calculated as an average of local dissimilarities between distance transforms of two given inputs. We could also without any problem use the thresholded definitions of the distances (which give rise to some Δ^* metrics). In this case, we have an upper bound of the dissimilarity between two images.

Property 1 (Uniform boundedness of the Δ^* metrics). *Given a distance transform based on a distance d , a threshold c and an integer q , the Δ^* metric defined as*

$$\forall f, g \quad \Delta_d^{q*}(f, g) = \left[\frac{1}{N_X} \sum_{x \in X} |DT^*[f](x) - DT^*[g](x)|^q \right]^{1/q} \quad (6)$$

is bounded by c , independently from the two inputs f and g .

Remark 1: if we do not add the thresholding operation to the distance would also get a bounded distance but the bound would depend on the size of the raster considered.

Remark 2: using $c > \dim(X)$ is equivalent to not using any thresholding at all.

Proof. Let's consider a distance measure between to spatial/spatial-intensity point in X d and a threshold value $c \geq 0$. Then we have $\forall p, p' \in X, d^*(p, p') \leq c$.

This implies that for any 2 given signals f and g , we have

$$0 \leq DT^*[f](p) \leq c \text{ and } 0 \leq DT^*[g](p) \leq c.$$

Therefore, we get $\forall p \in X, q \in \mathbb{N}, |DT^*[f](p) - DT^*[g](p)|^q \leq c^q$.

Finally, averaging the last expression and taking the q -root of it shows the result. \square

4.2 Experiments

In this section, we study the behavior of the different proposed approaches regarding small controlled perturbations. We will independently consider measuring dissimilarities between an image and its translated, rotated, or blurred variants. As a first experiment, we decided to stick to the use of the mansion image as already presented, as it is an example of natural image scene without too many complex structure, texture or background.

For comparison purposes, we have added two other metrics which are not entering the category of distance transform based measures but are still computing some structural (dis-)similarities between two images.

The first one is the SSIM index introduced by Wang *et al.* in 2004 [11]. This structural similarity index is computed by combining three separate level of information between images. A first similarity is computed to compare the intensities, then another one takes into account the difference of contrast between the images. Finally a last similarity is computed on the resulting structures. The three components are then combined together to build a local similarity measure between 0 and 1 where 1 denotes a perfect similarity and 0 and perfect mismatch.

The other chosen similarity for comparison is the one introduced by Moser in [10]. The discrepancy norm is a way of finding the biggest interval (in 1D cases) or convex sets/rectangles (in nD cases) with the biggest dissimilarity to the uniform 0 distribution. In other terms, computing the discrepancy norm of the difference of two signals or images is equivalent to finding the biggest dissimilarity or discrepancy on the biggest convex subset of the domain. This yields dissimilarity values which are positive, unbounded with 0 being the best match achievable and actually only achieved when measuring differences between two exact same signals (due to its norm property). It has been shown to be monotonic with respect to shiftings of a signal when computing autocorrelation values.

A last similarity value called *Local dissimilarity maps (LDMap)* [2] have been added to the comparison tables/plots. It is calculated by taking the pointwise product of the difference of the two images on one size, with the difference of the two distance transforms on the other:

$$LDMap(f, g)^q = \left\{ \frac{1}{N_X} \sum_{x \in X} (|f(x) - g(x)| \cdot |DT[f](x) - DT[g](x)|)^q \right\}^{1/q} \quad (7)$$

Dissimilarities and blurring For this purposes, the input image is blurred many times consecutively with a 5×5 gaussian blurring masking with variance $\sigma_{blur} = 0.5$



(a) House image after 10 blurring (b) House image after 20 blurring (c) House image after 35 blurring (d) House image after 50 blurring

Figure 5: Examples of blurred images used for analysis of the different measures.

| Blurring | MSE | Wilson | Molchanov | LDMap | SSIM | DN |
|----------|---------|--------|-----------|--------|--------|----------|
| 10 times | 12.6983 | 0.1029 | 0.7613 | 0.1783 | 0.8397 | 397.5537 |
| 20 times | 17.1617 | 0.1128 | 0.8802 | 0.2566 | 0.7776 | 660.1263 |
| 35 times | 21.3560 | 0.1273 | 1.0009 | 0.3346 | 0.7373 | 962.3175 |
| 50 times | 24.2942 | 0.1366 | 1.0728 | 0.3915 | 0.6846 | 1208 |

Table 1: Results of the different metrics on the shown samples of blurred images

One sees on the images that the local structural information are getting blurred away. Therefore we expect the dissimilarity values to follow this trend: the less structural proximity the two images have the highest should be the value (or smaller for the SSIM similarity index). Table 1 shows the values obtained on the images shown in Fig. 5.

One important remark to be done is that all of the proposed approaches show the expected behaviour. However, one should be careful when analyzing the results as different metrics have different dynamics. Because of that, we will not display the values for the discrepancy norm in plots (it shows much too high values and would blows comprehension away).

Fig.6 shows the behaviour of the different similarity measures. It seems that they all show quite a steep increase at the beginning and then tend to stabilize (after about 20 gaussian smoothing).

Dissimilarities and translation In this case, we have decided to expand our input image with zero padding around it so that no information is getting lost during the shifting process. For this purpose we have shifted the mansion image to the left, one pixel after another.

Once again, we have computed our different measures on those examples and gathered the results regarding the images in Fig. 7 in Table 2 and in Fig.8

Interestingly all the presented metrics show a rather nice monotonicity behaviour. It seems that

| Translation | MSE | Wilson | Molchanov | LDMap | SSIM | DN |
|-------------|--------|--------|-----------|--------|--------|------|
| 5px | 4.6562 | 1.4037 | 1.0869 | 0.5107 | 0.9399 | 900 |
| 10px | 6.2390 | 2.4323 | 1.8769 | 1.1477 | 0.9296 | 1794 |
| 20px | 8.3279 | 3.3104 | 2.5426 | 1.7079 | 0.9190 | 3570 |
| 30px | 9.7284 | 3.7842 | 2.8999 | 1.9897 | 0.9114 | 5349 |

Table 2: Results of the different metrics on the shown samples of translated images

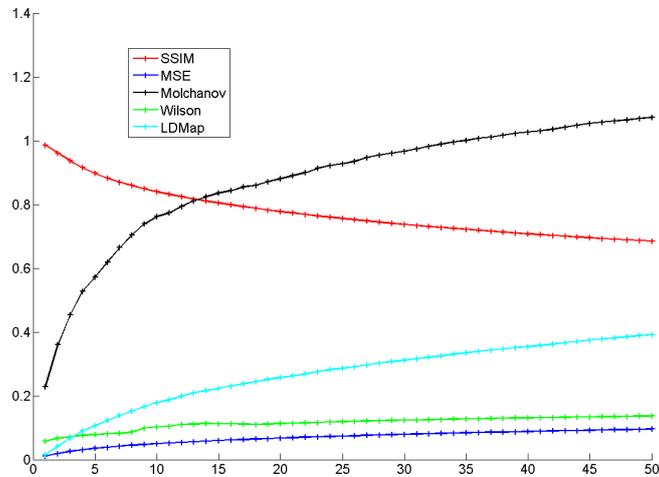


Figure 6: Similarities with increasing gaussian blur.

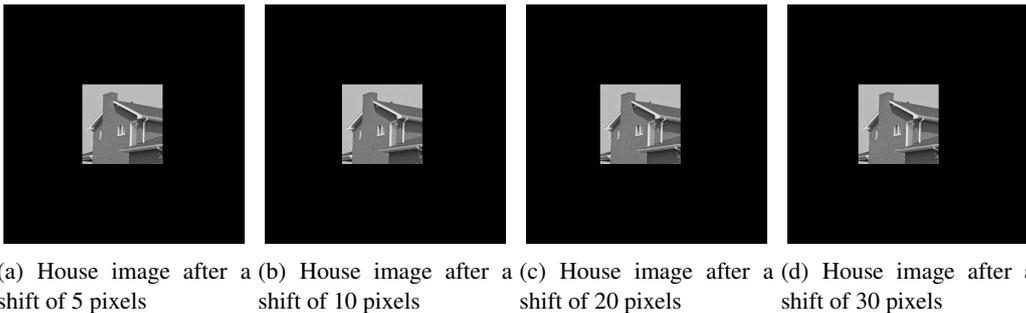


Figure 7: Examples of translated images used for analysis of the different measures.

considering structural information for a similarity measure is particularly important when dealing with misaligned images.

Dissimilarities and rotation In this case, we have decided to expand our input image with zero padding around it so that no information is getting lost during the rotation process. The input Mansion image is being rotated and the (dis-)similarities between the original image and its rotated variants is computed for all the proposed metrics. Some sample images are shown on Fig. 9.

The results for these images as well as other particular cases are given in Table 3. Results on a complete for all 90 rotations are gathered in Fig. 10

It appears that all the distance transform based metrics show some kind of cyclic behaviour by first growing until reaching a local maxima at around $40\text{--}50^\circ$ and then get down again. Our feeling is that the behaviour is mainly due to the zero padding artifacts which creates artificially high distances at each corner.

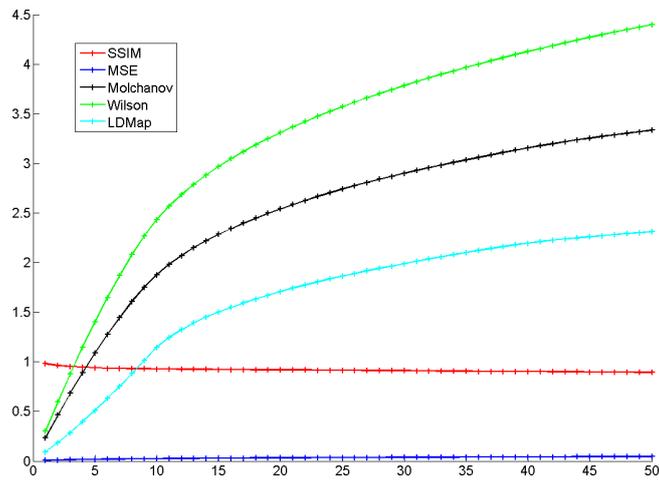
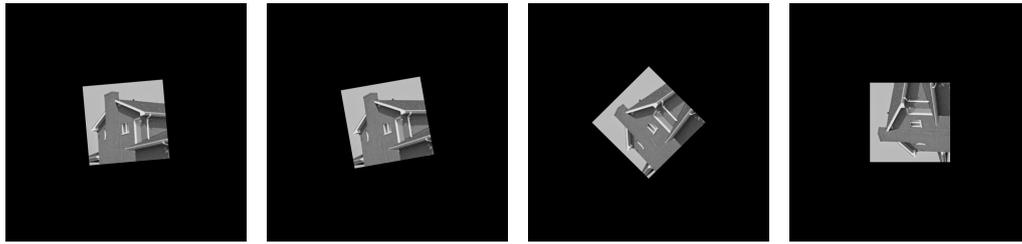


Figure 8: Similarities with increasing shift.

| Rotations | MSE | Wilson | Molchanov | LDMaP | SSIM | DN |
|-----------|---------|--------|-----------|--------|--------|------|
| 5 | 5.7108 | 2.1877 | 1.6454 | 0.9073 | 0.9342 | 1563 |
| 10 | 7.4848 | 3.0454 | 2.2986 | 1.4590 | 0.9217 | 2909 |
| 20 | 9.4796 | 3.7495 | 2.7887 | 1.8516 | 0.9069 | 4986 |
| 30 | 10.5129 | 4.0604 | 3.0413 | 2.0485 | 0.8987 | 6151 |
| 45 | 11.0306 | 4.2099 | 3.1517 | 2.1332 | 0.8948 | 7261 |
| 90 | 6.8741 | 2.4890 | 1.7951 | 0.7848 | 0.9240 | 5110 |

Table 3: Results of the different metrics on the samples of rotated images



(a) House image after a rotation of 5° (b) House image after a rotation of 10° (c) House image after a rotation of 45° (d) House image after a rotation of 90°

Figure 9: Examples of rotated images used for analysis of the different measures.

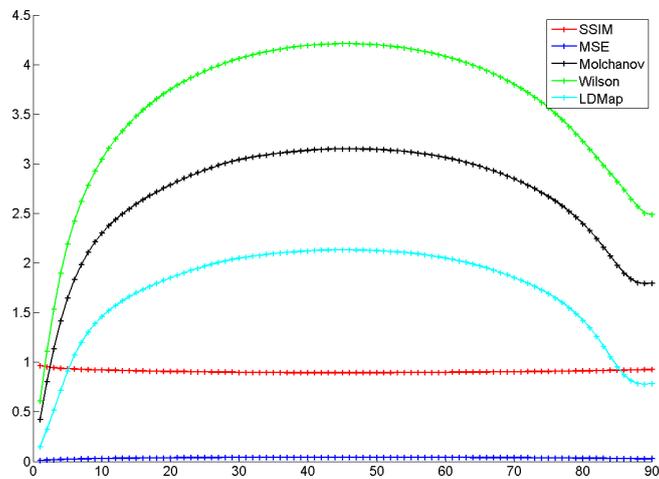


Figure 10: Similarities with increasing rotation angle.

5 Conclusion

We presented some results on our first researches on the topic of gray-level distance transforms for image comparisons. It is interesting that tuning the different parameters allows to give more importance to different level or fineness of details.

References

- [1] BADDELEY, A. An error metric for binary images. *Robust computer vision* (1992), 59–78.
- [2] BAUDRIER, E., MILLON, G., NICOLIER, F., AND RUAN, S. The adaptative local hausdorff-distance map as a new dissimilarity measure. *Pattern Recognition* 41 (2004), 1461–1478.
- [3] BORGEFORS, G. Distance transformations in digital images. *Computer vision, graphics, and image processing* 34, 3 (1986), 344–371.

-
- [4] BORGEFORS, G. On digital distance transforms in three dimensions. *Computer Vision and Image Understanding* 64, 3 (1996), 368–376.
 - [5] COQUIN, D., AND BOLON, P. Application of baddeley’s distance to dissimilarity measurement between gray scale images. *Pattern Recognition Letters* 22, 14 (2001), 1483–1502.
 - [6] FELZENSZWALB, P. F., AND HUTTENLOCHER, D. P. Distance transforms of sampled functions. Tech. rep., Cornell Computing and Information Science, 2004.
 - [7] FOUARD, C., AND GEDDA, M. An objective comparison between gray weighted distance transforms and weighted distance transforms on curved spaces. In *Discrete Geometry for Computer Imagery* (2006), Springer, pp. 259–270.
 - [8] MOLCHANOV, I., AND TERAN, P. Distance transforms for real-valued functions. *Journal of mathematical analysis and applications* 278, 2 (2003), 472–484.
 - [9] MONTANARI, U. A method for obtaining skeletons using a quasi-euclidean distance. *Journal of the ACM (JACM)* 15, 4 (1968), 600–624.
 - [10] MOSER, B. A similarity measure for image and volumetric data based on hermann weyl’s discrepancy. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 99 (2011), 1–1.
 - [11] WANG, Z., BOVIK, A., SHEIKH, H., AND SIMONCELLI, E. Image quality assessment: From error visibility to structural similarity. *Image Processing, IEEE Transactions on* 13, 4 (2004), 600–612.
 - [12] WILSON, D., BADDELEY, A., AND OWENS, R. A new metric for grey-scale image comparison. *International Journal of Computer Vision* 24, 1 (1997), 5–17.