Contrast Invariant SNR and Isotonic Regressions

Pierre Weiss, Paul Escande, Gabriel Bathie, Yiqiu Dong

Noname manuscript No.
(will be inserted by the editor)

Contrast Invariant SNR and Isotonic Regressions

Pierre Weiss, Paul Escande, Gabriel Bathie and Yiqiu Dong

the date of receipt and acceptance should be inserted later

Abstract We design an image quality measure independent of contrast changes, which are defined as a set of transformations preserving an order between the level lines of an image. This problem can be expressed as an isotonic regression problem. Depending on the definition of a level line, the partial order between adjacent regions can be defined through chains, polytrees or directed acyclic graphs. We provide a few analytic properties of the minimizers and design original optimization procedures together with a full complexity analysis. The methods worst case complexities range from $O(n)$ for chains, to $O(n \log n)$ for polytrees and $O\left(\frac{n^2}{\sqrt{\epsilon}}\right)$ for directed acyclic graphs, where n is the number of pixels and ϵ is a relative precision. The proposed algorithms have potential applications in change detection, stereo-vision, image registration, color image processing or image fusion. A C++ implementation with Matlab headers is available at `https://github.com/pierre-weiss/contrast_invariant_snr`.

Keywords Local contrast change, topographic map, isotonic regression, convex optimization, illumination invariance, signal-to-noise-ratio, image quality measure, dynamic programming.

P. Weiss is with ITAV and IMT, CNRS and Université de Toulouse, France.

P. Escande is with the DISC department, at ISAE, Université de Toulouse, France.

G. Bathie is a bachelor student at INSA de Toulouse, Université de Toulouse, France.

Y. Dong is with the department of applied mathematics and computer science at Technical University of Denmark.

Address(es) of author(s) should be given

1 Introduction

Invariance to illumination conditions is often a key element for the success of image processing algorithms. The whole field of mathematical morphology is based on contrast invariance [1]. The structural similarity index [2] - one of the most popular image quality measures - also strongly relies on a partial invariance to illumination changes.

1.1 Contrast invariant SNR

The main objective of this work is to measure the similarity between two images $u_0 : \Omega \to \mathbb{R}$ and $u_1 : \Omega \to \mathbb{R}$, where Ω is a discrete domain, in a way robust to illumination changes. To this end, we propose to solve the following variational problem:

$$\Delta_{\mathcal{T}}(u_1, u_0) = \min_{T \in \mathcal{T}} \|u_0 - T(u_1)\|_2^2, \tag{1}$$

where $\mathcal{T}$ is a family of transforms modeling illumination changes. Designing a family $\mathcal{T}$ reproducing faithfully variations of illuminations is probably out of reach and would most likely turn out to be of little interest from a computational point of view. Here, we will focus on simple families $\mathcal{T}$ that preserve the *level-lines* of the image u_1 as well as a partial order between them. This amounts to comparing images independently of *local contrast changes*, as defined in [3]. The contrast invariant signal-to-noise-ratio is then defined by:

$$SNR_{\mathcal{T}}(u_1, u_0) = -10 \log_{10}(\Delta_{\mathcal{T}}(u_1, u_0)/\|u_0\|_2^2). \tag{2}$$

1.2 Potential applications

In this paper, we primarily focus on the basic properties of the model and exemplify its use as an image qual-

ity metric. It can however be used in more advanced applications mentioned below.

Change detection Let $T^\star$ denote the optimal transform in problem (1). The image $u^\star = T^\star(u_1)$ has the same level-lines (i.e. geometry) as u_1, with the contrast of u_0. The difference image $u_0 - T^\star(u_1)$ contains the "objects" in u_0 which are not in u_1. The proposed tool can therefore be used as a contrast invariant change detection algorithm. We show a few difference images in Fig. 6. Let us mention that a similar idea was proposed in [4] and [5]. It was however based on a purely morphological approach, while the proposed method is both morphological and variational.

Inverse problems The image $T^\star(u_1)$ is the projection of u_0 on a convex set. Projections and more generally proximal operators have proved to be key tools for the resolution of inverse problems [6,7]. The proposed algorithms may therefore be used as a basic brick in more advanced inverse problems, where a natural regularizer is to impose a given topographic map. The proposed method hence possesses potential applications in different fields such as image fusion [8] or color image processing [8–10].

Let us exemplify how it could be used for panchromatic and multi-spectral image fusion. In this application, we are given a high resolution image u_0 and a set of low resolution images v_i at different wavelengths. The aim is to construct a high resolution multi-spectral image. If the wavelengths of u_i are not too far from the wavelength of u_0, we may assume that no local contrast inversion occur. Fusing u_0 and an image u_i might therefore be achieved by solving:

$$\min_{u \in \mathcal{T}(u_0)} \frac{1}{2}\|Hu - u_i\|_2^2,$$

where H is a down-sampling operator and where $\mathcal{T}(u_0)$ is the set of admissible images.

Image registration Given two images u_1 and u_0 coming from different modalities, or from the same modality with different stainings, we may try to register them by solving a problem of the form:

$$\inf_{d \in \mathcal{D}} \Delta_{\mathcal{T}}(u_1 \circ d, u_0), \tag{3}$$

where $\mathcal{D}$ is a family of admissible deformations and $\Delta_{\mathcal{T}}$ is one of the measures proposed in the paper. This idea is similar to an idea developed in [11], with the important difference that we promote level-lines with an identical partial order, while [11] only focussed on aligning level-lines, with no care for their relative orders.

Optical flow The same idea can be applied to estimate the optical flow, where changes of illuminations need to be taken with care (see e.g. section 6.3.3 of [12]). Under the assumption that the deformation d in equation (3) is small, we may linearize $u_1 \circ d$ as $u_1 \circ d \simeq u_1 + \nabla u_1 \odot v$, where v is a displacement field of small amplitude and where $\nabla u_1 \odot v$ is the image corresponding to the pixel-wise scalar product between v and ∇u_1. This simplification leads to problems of the form:

$$\inf_{v \in \mathcal{V}} \Delta(u_1 + \nabla u_1 \odot v, u_0), \tag{4}$$

where $\mathcal{V}$ is a set of admissible displacements. If the set $\mathcal{V}$ is convex, the problem (4) is convex too and could be solved efficiently using first-order methods (see e.g. [12]) combined with the tools provided later in this paper. We leave the practical investigation of this idea for future works.

1.3 Isotonic regressions

After a few algebraic manipulations, the variational problem (1) can be turned into an *isotonic regression* problem with a structure that depends on the family of transforms $\mathcal{T}$. Isotonic regressions have been introduced in the context of statistics in the 1950's [13]. Letting (V, E) denote the vertices and edges of a directed acyclic graph (DAG) and $y \in \mathbb{R}^{|V|}$ denote a label of the graph, they can be written as follows:

$$\min_{\substack{x \in \mathbb{R}^{|V|} \\ x_j - x_i \geq 0, \forall (i,j) \in E}} d(x, y), \tag{5}$$

where $d : \mathbb{R}^{|V|} \times \mathbb{R}^{|V|} \to \mathbb{R}$ is a convex function measuring the closeness from x to y.

Solving (5) efficiently is a rather involved problem despite its convexity. It has received a considerable attention and significant progresses have been achieved recently [14–20]. The existing methods may be exact or approximate and their complexity depends on the structure of the directed acyclic graph. In our work, we will consider 3 different types of graphs: i) linear graphs with a single orientation, ii) polytrees and iii) generic directed acyclic graphs.

The case of linear graphs with a single orientation is well understood and can be solved in $O(|V|)$ operations using pool adjacent violators algorithms for instance [16].

To the best of our knowledge, the case of polytrees has not been studied in the literature yet, even though the case of directed trees is well understood [17]. In a recent work, Kolmogorov, Pock and Rolinek [21] introduced efficient dynamic programming approaches

to solve families of convex problems defined on trees. Building upon this work, we design algorithms with a complexity $O(|V|)$ when the graph is linear with edges oriented in arbitrary directions. We also design an algorithm with complexity $O(|V|\log|V|)$, when the graph is a polytree (i.e. a tree with edges in arbitrary orientations). To the best of our knowledge, this is the first time that these problems are considered. Combining these tools with the Fast Level Set Transform [22] of Guichard and Monasse, we can solve an instance of our main problem (1) with a worst-case complexity $O(n\log n)$, where n is the number of pixels in the image.

Finally, we introduce a simple first order algorithm in the case of arbitrary directed acyclic graphs, together with a full complexity analysis. The proposed method has some comparative advantages with the current state-of-the-art approaches [18,20] both from a theoretical and practical point of view.

2 Existing approaches

Various approaches are commonly used to compare two images u_0 and u_1 independently of illumination variations. We briefly describe a few of them below.

2.1 Contrast equalization

Probably the most common approach consists in equalizing histograms, i.e. to change the gray-values of u_1 in such a way that the resulting histogram matches approximately that of u_0 (see e.g. [23, 24]). This approach suffers from the fact that the image geometry is completely forgotten: histograms only account for gray-level distributions and not geometrical features such as edges, textures,...

2.2 Mutual information

Mutual information has been popularized in the field of image registration [25, 26] to compare images. In its simplest form, the mutual information treats the gray values of two images u_0 and u_1 as random variables X and Y with values in a discrete set Λ. This measure is then defined as:

$$H(X,Y) = \sum_{x\in\Lambda, y\in\Lambda} P(x,y)\log\left(\frac{P(x,y)}{P(x)P(y)}\right),$$

where $P(x,y)$ denotes the probability of the event $X = x$ and $Y = y$, $P(x)$ the probability of the event $X = x$ and $P(y)$ the probability of the event $Y = y$.

Mutual information is based solely on the gray level distributions and does not account for the image geometry (that could be captured by the gradient for instance). It is invariant to affine transforms of the pixels values, but not to more complex nonlinear mappings of the gray levels. More generally, the measure is not invariant to local contrast changes as defined in this paper (i.e. changes that can affect distant pixels in a different manner). In addition, the mutual information $H(X,X)$ of two identical images is equal to the entropy of X which varies from an image to another, while we could expect from a metric to yield an identical value.

2.3 Optimal linear and affine maps

The set $\mathcal{T}$ in problem (1) can be replaced by any class of transformations that describe changes of illuminations. Probably the simplest classes $\mathcal{T}$ are the set of linear maps $T(u) = au$ or the set of affine maps $T(u) = au+b$, where a and b are scalars. The solution of both problems can be computed explicitly in terms of u_0 and u_1. The same approach can be used locally and the L^2-norm can be replaced by a weighted L^2-norm. This idea is the basis of the Structural Similarity Index Measure (SSIM).

2.4 Optimal global contrast change

A richer set of transformations $\mathcal{T}$ is that of global contrast changes. Two images u_0 and u_1 are said to differ by a global contrast change if there exists a non decreasing function $T:\mathbb{R}\to\mathbb{R}$ such that $T\circ u_1 = u_0$, where the composition $T\circ u_1$ has to be understood pixelwise. Let

$$\mathcal{U}_{glo} = \{u = T\circ u_1, T:\mathbb{R}\to\mathbb{R},\ \text{non decreasing}\} \quad (6)$$

Finding the best global contrast change amounts to solving:

$$\Delta_{glo}(u_1,u_0) = \min_{u\in\mathcal{U}_{glo}} \frac{1}{2}\|u-u_0\|_2^2. \quad (7)$$

We let $u^\star_{glo}$ denote the solution of (7) and

$$SNR_{glo}(u_1,u_0) = -10\log_{10}(\Delta_{glo}(u_1,u_0)/\|u_0\|_2^2)$$

denote the globally contrast invariant SNR. This approach seems to be vastly ignored. In fact we only found its description in the excellent lecture notes of Lionel Moisan [27] and we will therefore quickly recall its principle below.

In what follows, we let

$$\mathcal{U}_{glo} = \{u = T\circ u_1, T:\mathbb{R}\to\mathbb{R},\ \text{non decreasing}\}. \quad (8)$$

and $\Lambda = (\lambda_i)$ denote the set of gray-scale values $u_1(\Omega)$ sorted in ascending order. The infinite dimensional problem (7) can be turned to an optimization of a vector $\alpha \in \mathbb{R}^{|\Lambda|}$. We first define the discrete level lines of u_1 by $\rho_i = \{x \in \Omega,\, u_1(x) = \lambda_i\}$ and get

$$\sum_{x\in\Omega} (T(u_1(x)) - u_0(x))^2 = \sum_{1\leq i\leq|\Lambda|} \sum_{x\in\rho_i} (T(\lambda_i) - u_0(x))^2.$$

Letting

$$\mathcal{S} = \left\{\alpha \in \mathbb{R}^{|\Lambda|},\, \alpha_{i+1} - \alpha_i \geq 0, \quad \forall 1 \leq i < |\Lambda|\right\},$$

problem (7) then becomes:

$$\min_{\alpha\in\mathcal{S}} \sum_{1\leq i<|\Lambda|} \sum_{x\in\rho_i} (\alpha_i - u_0(x))^2, \tag{9}$$

where $\alpha_i = T(\lambda_i)$. We will use the following result repeatedly in the paper:

Lemma 1 *Let $\beta_i = \bar{u}_0(\rho_j) := \frac{1}{|\rho_i|}\sum_{x\in\rho_i} u_0(x)$ be the mean of u_0 over the region ρ_i. We have for all $\alpha_i \in \mathbb{R}$:*

$$\sum_{x\in\rho_i} (\alpha_i - u_0(x))^2 = |\rho_i|\mathrm{Var}_{\rho_i}(u_0) + |\rho_i|(\alpha_i - \beta_i)^2, \tag{10}$$

where $\mathrm{Var}_{\rho_i}(u_0) := \frac{1}{|\rho_i|}\sum_{x\in\rho_i}(\beta_i - u_0(x))^2$ and where $|\rho_j|$ denotes the cardinality of ρ_j.

Proof

$$\begin{aligned}
&\sum_{x\in\rho_i} (u_0(x) - \alpha_i)^2 \\
&= \sum_{x\in\rho_i} (u_0(x) - \beta_i + \beta_i - \alpha_i)^2 \\
&= \sum_{x\in\rho_i} (u_0(x) - \beta_i)^2 + (\beta_i - \alpha_i)^2 + 2(u_0(x) - \beta_i)(\beta_i - \alpha_i) \\
&= |\rho_i|\mathrm{Var}_{\rho_i}(u_0) + |\rho_i|(\beta_i - \alpha_i)^2.
\end{aligned}$$

Therefore Problem (7) can be rewritten as:

$$\min_{\alpha\in\mathcal{S}} \sum_{1\leq i<|\Lambda|} |\rho_i|(\alpha_i - \beta_i)^2 + |\rho_j|\mathrm{Var}_{\rho_j}(u_0), \tag{11}$$

The problem (11) is a simple case of isotonic regression [15,16] on a linear graph. This problem can be solved in $O(n)$ operations using active sets type methods called pool adjacent violators. Unfortunately, global contrast changes do not capture all the complexity of illumination changes: in most applications, the variations are local and we will therefore introduce more complex sets of transformations in the next section.

3 Local contrast changes using a tree

3.1 The tree of shapes

Mathematical morphology emerged with the works of Matheron [28]. Therein, he proposed to analyze and process an image u by operating on their upper level-sets $\chi^+_\lambda = \{x \in \Omega, u(x) \geq \lambda\}$ and lower level-sets $\chi^-_\lambda = \{x \in \Omega, u(x) < \lambda\}$. The level-sets are geometrical features invariant to global contrast changes. In order to obtain a representation invariant to *local* contrast changes, it is possible to consider their saturated [1] connected components called *shapes*. This idea was proposed and detailed thoroughly in [3, 29] in the continuous setting and in [22, 30] in the discrete one.

Let $(\omega_i)_{i\in I}$ denote the set of shapes, i.e. the set of saturated connected components of the lower and upper level-sets. In the discrete setting and under suitable choices of connexity [2], it can be shown that the family of shapes $(\omega_i)_{i\in I}$ satisfy the following inclusion relationships:

$$\text{For } i \neq j, \text{either } \omega_i \subset \omega_j \text{ or } \omega_j \subset \omega_i \text{ or } \omega_j \cap \omega_i = \emptyset. \tag{12}$$

This allows to embed the shapes in a tree called the tree of shapes. This idea goes back to Kronrod [31] and we refer the interested reader to the book [29] for more insight on its properties in the continuous setting. The vertices of the tree coincide with the shapes. A shape ω_j is a descendant of ω_i in the tree if $\omega_j \subset \omega_i$. This property allows to define the set of edges E, encoding the inclusion relationship between shapes. An edge e is a pair of indices (i, j) indicating that ω_j is a child of ω_i. An illustration of this idea is proposed in Fig. 1. In this figure, we added arrows to the tree, indicating whether the difference of gray values between consecutive shapes is positive or negative.

In the discrete setting and for 2D images, the tree of shapes can be constructed in $O(n \log n)$ operations using the so-called fast level-set transform (FLST) [22]. In arbitrary dimensions, the complexity remains moderate if the images are quantized [30].

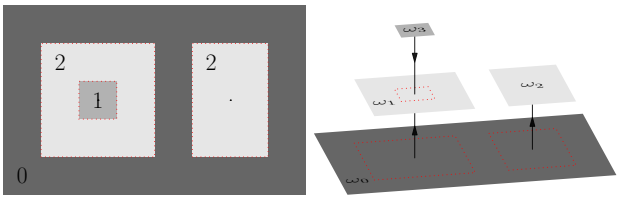
3.2 Local contrast changes on a tree

The level-set transform can be seen as a decomposition of the image u of the following form:

$$u = L\gamma := \sum_{i\in I} \gamma_i s_i \mathbb{1}_{\omega_i}, \tag{13}$$

[1] The saturation of a set S is constructed by filling the holes of S. A hole is defined as a connected component of the complementary of S which is in the interior of S.

[2] One should choose the 4 connexity for the upper-level sets and the 8 connexity for the lower level-sets (or the reverse) to satisfy a discrete version of Jordan's theorem [22]



(a) An image (b) Its tree of shapes

Fig. 1: An example of tree of shapes

where I is a set of cardinality $|I| = |V| \leq n$, $(\gamma_i) \in \mathbb{R}_+^{|I|}$ is a vector of nonnegative coefficients encoding the jump between a shape and its parent and $s_i \in \{-1, 1\}$ is a sign vector indicating whether ω_i is a connected component of a lower or upper level-set of u. Unfortunately, the decomposition of an image using the linear mapping $L : \gamma \to u$ in equation (13) is not computationally attractive. Indeed, if L is stored as a sparse matrix, it can contain up to $O(n^2)$ nonzero coefficients since a single pixel can belong to $O(n)$ shapes.

A more efficient reconstruction formula requires using the *level-lines* instead of the level-sets.

Definition 1 (Level-lines on a tree) Let children(i) denote the indices of the children of ω_i. The i-th level-line $\partial\omega_i$ is defined by $\partial\omega_i = \omega_i \backslash (\cup_{j \in \text{children}(i)} \omega_j)$[3].

The set of level-lines $(\partial\omega_i)_{i\in I}$ forms a partition of the image domain Ω and we have the following decomposition formula:

$$u = R\alpha := \sum_{i \in I} \alpha_i \mathbb{1}_{\partial\omega_i}. \tag{14}$$

where $\alpha_i = u|_{\partial\omega_i}$ is the value of u restricted to the level-line. A matrix-vector product with R has a complexity $O(n)$ since the level-lines partition Ω. The relationship between α and γ is simply $\alpha_j - \alpha_i = \gamma_j s_j$, for all $(i, j) \in E$ since γ defines the jump between adjacent level-lines. We can now introduce the first definition of a local contrast change.

Definition 2 (Local contrast change on a tree) Let u_1 denote a reference image, (E, V) denote its tree of shapes and R denote its associated reconstruction operator. An image u is said to differ from u_1 by a local contrast change on a tree (LCCT) if it can be written as $u = R\alpha$, where $(\alpha_j - \alpha_i)s_j \geq 0$ for all $(i, j) \in E$.

In words, this means that u_1 and u should have the same level-lines and that the difference between the gray levels of connected level-lines should have the same sign.

[3] This is a slight abuse of notation since a level-line defined this way can have a nonempty interior.

We are now ready to convert the problem (1) into an isotonic regression problem. As in the previous section, we let

$$\mathcal{U}_{loc1} = \{u \text{ differs from } u_1 \text{ by a LCCT}\}, \tag{15}$$

$$\Delta_{loc1}(u_1, u_0) = \min_{u \in \mathcal{U}_{loc1}} \frac{1}{2}\|u - u_0\|_2^2,$$

$$SNR_{loc1}(u_1, u_0) = -10 \log_{10}\left(\frac{\Delta_{loc1}(u_1, u_0)}{\|u_0\|_2^2}\right),$$

and $u^\star_{loc1}$ denote the solution of (15). Letting $\Lambda = \{\alpha \in \mathbb{R}^{|I|}, (\alpha_j - \alpha_i)s_j \geq 0, \forall (i, j) \in E\}$, we get the following problem:

$$\min_{u \in \mathcal{U}_{loc1}} \frac{1}{2}\|u - u_0\|_2^2 \tag{16}$$

$$= \min_{\alpha \in \Lambda} \frac{1}{2}\|R\alpha - u_0\|_2^2 \tag{17}$$

$$= \min_{\alpha \in \Lambda} \frac{1}{2}\|\sum_{i \in I} \alpha_i \mathbb{1}_{\partial\omega_i} - u_0\|_2^2 \tag{18}$$

$$= \min_{\alpha \in \Lambda} \frac{1}{2}|\partial\omega_i|\text{Var}_{\partial\omega_i}(u_0) + \frac{1}{2}\sum_{i \in I} |\partial\omega_i|(\alpha_i - \beta_i)^2, \tag{19}$$

where we used Lemma 1 with $\beta_i := \bar{u}_0(\partial\omega_i)$. By skipping the constant terms $\frac{1}{2}|\partial\omega_i|\text{Var}_{\partial\omega_i}(u_0)$, this problem can still be rewritten as:

$$\min_{\alpha \in \mathbb{R}^{|I|}} \sum_{i \in I} f_i(\alpha_i) + \sum_{(i,j) \in E} f_{i,j}(\alpha_j - \alpha_i), \tag{20}$$

with $f_i(\alpha_i) = \frac{1}{2}|\partial\omega_i|(\alpha_i - \beta_i)^2$ and

$$f_{i,j}(z) = \begin{cases} 0 & \text{if } zs_j \geq 0 \\ +\infty & \text{otherwise.} \end{cases} \tag{21}$$

The equation (20) corresponds to an isotonic regression problem on a polytree, for which we will design an efficient algorithm in the next section.

3.3 A dynamic programming approach

The review paper [32] shows that problems of the form (20) can be solved efficiently using dynamic programming, when the values of α are restricted to a set of finite cardinality. In [21], the authors show that this restriction is not necessary provided that the functions f_i and $f_{i,j}$ have a favorable convex structure. They give a particular attention to combinations of piecewise quadratic and piecewise linear functions. This is very close to our setting since f_i is quadratic and that the function $f_{i,j}$ in (21) can be seen as a degenerate piecewise linear function with an infinite slope. We sketch the principle of the approach below.

We let $M_i(x)$ denote the optimal value of the energy restricted to the subtree rooted at i, and fixing the value x at node i. This energy function can be computed directly on the leaves since it only contains the unary terms f_i. It can then be propagated to the parents by using the recursion formula:

$$M_i(x) = f_i(x) + \sum_{j \in \text{children}(i)} (M_j \square f_{i,j})(x), \tag{22}$$

where the symbol $\square$ stands for the inf-convolution defined by:

$$(M_j \square f_{i,j})(x) = \inf_{z \in \mathbb{R}} M_j(z) + f_{i,j}(x - z). \tag{23}$$

The dynamic programming approach works by first evaluating recursively the functions M_i on each node in a first pass from the leaves towards the root. Then, the optimal solution can be obtained using a backpropagation: the optimal value $x_r^\star$ at the root r is obtained by $x_r^\star = \text{argmin}_{x \in \mathbb{R}} M_r(x)$. The rest of the values can be computed using the recursion:

$$x_j^\star = \operatorname*{argmin}_{x \in \mathbb{R}} M_j(x) + f_{i,j}(x - x_i), \tag{24}$$

for all $j \in \text{children}(i)$.

Making this general principle practical and efficient requires many subtelties which are well described in [21]. We review the main elements below:

- First, the messages M_i can be shown to be piecewise quadratic, with a number of pieces that does not exceed the number of nodes in the tree. Since the functions M_i are used only for minimization purposes, it is more practical to encode them through their subgradients m_i which are piecewise linear. Hence, they can be simply encoded as a finite sequence of slopes and breakpoints.
- In equation (22), the inf-convolution and the summation of the messages need to be computed in an efficient manner. In practice, this requires using an advanced data structure called double ended priority queue [33]. In our codes, we simply used a double ended queue from the standard C++ library, since we could find no open-source implementation allowing to do all the necessary operations (min, max, merge, removemin, removemax, insertmin, insertmax).
- The messages m_i are not stored on each node of the tree, since it would require storing up to $O(n^2)$ numbers. In practice it suffices to store the location of the minimum of M_i, which is the only information necessary in the back-propagation.

Using these tricks, it can be shown that the method's worst case complexity is $O(|I| \log |I|)$. Our current implementation has a worst case complexity in $O(|I|^2)$ since we did not use a double ended priority queue, but it turns out to have a near linear complexity in practice. This is due to the fact that the number of breakpoints does not scale as n for practical problems, but remains small compared to n. We illustrate the practical method's behavior in Fig. 2. The points on the curves represent the average computing time evaluated by comparing 16 pairs of natural images. As can be seen on the green dash-dotted curve, the behavior of the dynamic programming algorithm is slightly worst than the one of the tree of shapes for large image sizes, but is still very efficient. For instance, an image of size 1000×1000 is treated in 0.7 seconds on average using a single core of a personal laptop.

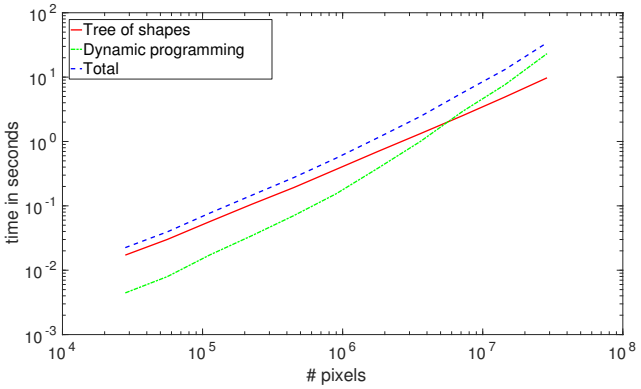


Fig. 2: Log-log plot of the computing times in seconds w.r.t. the number of pixels of the image. Red: time to construct the tree of shapes. Green: time to solve the isotonic regression via dynamic programming. Blue: total time for the algorithm (including copies of trees, arrays, etc...).

3.4 Comparison to the state-of-the-art

Nearly all the ideas described above are present in the paper [21]. The only difference is that we consider degenerate functions $f_{i,j}$ with an infinite slope, but the theoretical analysis remains unchanged. Our contribution on the side of dynamic programming is therefore mostly to point out that the dynamic programming approach from [21] allows to solve many instances of isotonic regressions using a single algorithm. Some of these instances were not considered previoulsy in the literature. We list the improvements below:

Fig. 3: Left: a simple image u_1. Right: a level-line in the tree of shapes has two disconnected components.

- We exactly solve isotonic regression problems on linear graphs with arbitrary orientations in $O(|I|)$ operations. This seems to be a novel result.
- We exactly solve isotonic regression problems on polytrees in $O(|I|\log|I|)$ operations. This seems to be a novel result.
- Using the algorithm in section 4.3 of [21], it can be shown that isotonic regressions with an ℓ^1 data fitting term on a linear graph can be solved in $O(|I|\log\log|I|)$ operations, while the best existing complexity was $O(|I|\log|I|)$ [34].

To finish, let us mention that our C++ implementation with a Matlab header is seemingly the first one to deal with isotonic regression on trees and polytrees.

4 Local contrast changes using a DAG

4.1 Local contrast changes using a DAG

The definition of level-lines and local contrast changes provided in the previous section has the important advantage of yielding a tree structure, which allows designing very efficient algorithms. In some cases, this definition might however be too global. This is illustrated in Fig. 3. In this picture, we see that two disconnected regions might be linked together when a level-set is separated by one of its children. In practice we may want to design an algorithm able to set different values to the left and right parts of the level-line. This is the purpose of this section.

4.1.1 Definitions

We endow the image domain Ω with a neighborhood relationship $\mathcal{N} : \Omega \to \mathcal{P}(\Omega)$. We assume that $\mathcal{N}$ is symmetric, meaning that $x \in \mathcal{N}(y) \Rightarrow y \in \mathcal{N}(x)$ for all $y \in \Omega$. We will focus on the 4-connectedness to establish theoretical properties of the model and in the numerical experiments. For each pixel $x \in \Omega$, the set $\mathcal{N}(x)$ is the set of all neighbors of x. We assume that $|\mathcal{N}(x)| \leq c_{\max}$ for all x, with $c_{\max} = 4$ when using the 4-connectedness denoted $\mathcal{N}_4$. We can now introduce a second definition of level-lines and local contrast changes.

Definition 3 (Level-lines on a DAG) The set of level-lines $(\Delta_i)_{1\leq i\leq p}$ of an image u_1 is the set of $\mathcal{N}$-connected component of $\{x \in \Omega, u_1(x) = \lambda\}$ for $\lambda \in u_1(\Omega)$.

The set of all level-lines $(\Delta_i)_{1\leq i\leq p}$ partitions the image domain Ω and has a cardinality $p \leq n$. In what follows, the notation $\Delta_i \overset{\mathcal{N}}{\sim} \Delta_j$ means that the level-lines are adjacent, i.e. that there exists $x \in \Delta_i$ and $y \in \Delta_j$ such that $y \in \mathcal{N}(x)$.

Definition 4 (Local contrast changes on a DAG) Let u and u_1 denote two images. The image u is said to differ from u_1 by a local contrast change on a DAG (LCCD) if $u|_{\Delta_i}$ is constant for all i and if the gray levels between adjacent pixels have the same order, i.e. for all $x \overset{\mathcal{N}}{\sim} y$:

$$(u(x) - u(y)) \cdot (u_1(x) - u_1(y)) \geq 0. \tag{25}$$

As in the previous section, we let

$$\mathcal{U}_{loc2} = \{u \text{ differs from } u_1 \text{ by a LCCD}\}, \tag{26}$$

$$\Delta_{loc2}(u_1, u_0) = \min_{u\in\mathcal{U}_{loc2}} \frac{1}{2}\|u - u_0\|_2^2,$$

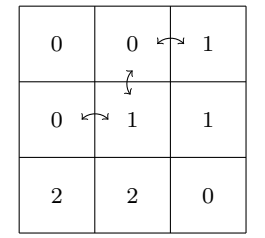
$$SNR_{loc2}(u_1, u_0) = -10\log_{10}\left(\frac{\Delta_{loc2}(u_1, u_0)}{\|u_0\|_2^2}\right),$$

and $u^\star_{loc2}$ denote the solution of (26).

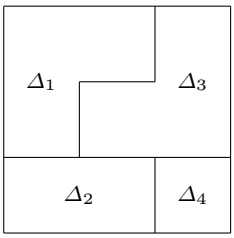
4.1.2 Constructing the graph

From an algorithmic point of view, we need to construct the sets $(\Delta_i)_{1\leq i\leq p}$ and a directed acyclic graph $G = (V, E)$ from the image u_1. The vertices $V = (v_1, \ldots, v_p)$ of this graph represent the sets $(\Delta_i)_{1\leq i\leq p}$. The set $E = (e_1, \ldots, e_m)$ contains the edges of the graph. The edge $e_k \in E$ is an ordered pair of vertices written $e_k = (\text{I}(k), \text{J}(k))$ going from vertex $\text{I}(k)$ to vertex $\text{J}(k)$. Such an edge exists if the sets $\Delta_{\text{I}(k)} \overset{\mathcal{N}}{\sim} \Delta_{\text{J}(k)}$ and if $u_1(\Delta_{\text{J}(k)}) > u_1(\Delta_{\text{I}(k)})$. The graph G can be encoded through an incidence matrix (or more precisely its transpose) $A \in \mathbb{R}^{m\times p}$. Each row of this matrix describes an edge with the convention $A(k, \text{I}(k)) = -1$ and $A(k, \text{J}(k)) = 1$ and all the other coefficients of row k are null. A simple 3×3 image u_1, the associated regions $(\Delta_i)_{1\leq i\leq 4}$, graph and incidence matrix are represented in Figure 4.

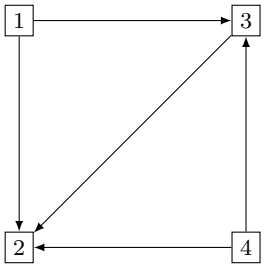
The list of regions $(\Delta_i)_{1\leq i\leq p}$ can be constructed in $O(n)$ operations using flood fill algorithms. This yields a labeling of the regions $(\Delta_i)_i$. The graph or matrix A can be constructed in $O(n\log n)$ operations. The idea is to first scan all the edges in $\mathcal{N}$ to construct a preliminary matrix $\tilde{A}$ with repetitions. For instance, the region Δ_1 is connected three times to Δ_3 (see arrows in Figure



(a) Image u_1



(b) Connected components



(c) Associated graph

$$A = \begin{pmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{pmatrix}$$

(d) Transpose of the incidence matrix

Fig. 4: Illustration of the graph construction

4a), so that after the first scan, the matrix $\tilde{A}$ contains three identical rows. The complexity of building this initial matrix is $O(c_{\max} n)$. Then, the repetitions can be suppressed in $O(n \log n)$ operations. First, the array can be rearranged by putting the identical rows adjacently (this can be achieved with a quicksort algorithm), then, the repetitions are removed.

4.1.3 The optimization problem

With these definitions, the problem (1) can be rephrased as follows:

$$\min_{u \in \mathcal{U}_{loc2}} \frac{1}{2}\|u_0 - u\|_2^2, \tag{27}$$

where $\mathcal{U}_{loc2}$ is the set of images satisfying Definition 4:

$$\mathcal{U}_{loc2} = \{u : \Omega \to \mathbb{R}, u|_{\Delta_i} = \alpha_i,\ 1 \leq i \leq p, A\alpha \geq 0\}.$$

Hence, we can simplify (27) as follows:

$$\begin{aligned} &\min_{u \in \mathcal{U}_{loc2}} \frac{1}{2} \sum_{x \in \Omega} (u_0(x) - u(x))^2 \\ &= \min_{u \in \mathcal{U}_{loc2}} \frac{1}{2} \sum_{i=1}^{p} \sum_{x \in \Delta_i} (u_0(x) - \alpha_i)^2 \\ &= \min_{\alpha \in \mathbb{R}^p, A\alpha \geq 0} \frac{1}{2} \sum_{i=1}^{p} |\Delta_i|(\beta_i - \alpha_i)^2 + |\Delta_i| \mathrm{Var}_{\Delta_i}(u_0), \end{aligned}$$

where $\beta_i = \bar{u}_0(\Delta_i)$. By letting $w \in \mathbb{R}^p$ denote the vector with components $w_i = |\Delta_i|$, $W = \mathrm{diag}(w)$ and by skipping the constant terms $|\Delta_i|\mathrm{Var}_{\Delta_i}(u_0)$, problem (1) finally simplifies to:

$$\min_{A\alpha \geq 0} \frac{1}{2}\langle W(\alpha - \beta), \alpha - \beta \rangle. \tag{28}$$

Problem (28) is an isotonic regression problem on a DAG. Its solution $\alpha^\star$ is unique since it is the projection of β onto a *closed convex set*. We will design a simple first order solver in the next section.

4.2 A first order solver

The main idea is to exploit the strong convexity of the squared l^2-norm to design a first order algorithm on the dual. Nesterov type accelerations [35] make this method particularly relevant for large scale problems [36]. Proposition 1 summarizes the nice properties of the dual problem.

Proposition 1 *The dual problem of* (28) *reads:*

$$\sup_{\lambda \leq 0} D(\lambda) = -\frac{1}{2}\|W^{-1/2}A^T\lambda\|_2^2 + \langle \lambda, A\beta \rangle. \tag{29}$$

Let $\alpha(\lambda) = \beta - W^{-1}A^T\lambda$, then any primal-dual solution $(\alpha^\star, \lambda^\star)$ satisfies $\alpha^\star = \alpha(\lambda^\star)$. The function D is differentiable with an L-Lipschitz continuous gradient and $L = \lambda_{\max}(AW^{-1}A^T)$. Finally, the following inequality holds for any $\lambda \in \mathbb{R}_-^m$ [4]*:*

$$\|\alpha(\lambda) - \alpha^\star\|_2^2 \leq 2(D(\lambda^\star) - D(\lambda)). \tag{30}$$

We refer the reader to the Appendix A for the proofs. In addition, the Lipschitz constant L can be bounded above by a constant.

Proposition 2 *The Lipschitz constant L satisfies $L \leq 4c_{\max}$.*

The problem (29) has a simple structure, compatible with the use of accelerated projected gradient ascents methods [37] described in Algorithm 1.

Algorithm 1 Accelerated proximal gradient ascent method.

1: **input:** initial guess $\mu^{(1)} \in \mathbb{R}^m$, $\tau = 1/4c_{\max}$ and Nit.
2: **for** $k = 1$ to Nit **do**
3: $\quad \lambda^{(k)} = \min\left(\mu^{(k)} + \tau \nabla D(\mu^{(k)}), 0\right)$.
4: $\quad \mu^{(k+1)} = \lambda^{(k)} + \frac{k-1}{k+2}(\lambda^{(k)} - \lambda^{(k-1)})$.
5: $\quad \alpha^{(k)} = \alpha(\lambda^{(k)})$.
6: **end for**

Proposition 3 *Algorithm 1 provides the following guarantees:*

$$\|\alpha^{(k)} - \alpha^\star\|_2^2 \leq \frac{8c_{\max}\|\lambda^{(0)} - \lambda^*\|_2^2}{(k+1)^2}, \tag{31}$$

[4] This result can be strengthened slightly, we refer the interested reader to the example 3.1 in [7] for more details.

where $\lambda^\star$ is any solution of the dual problem (29)*. The complexity to obtain an estimate $\alpha^{(k)}$ satisfying $\|\alpha^{(k)} - \alpha^\star\|_2 \leq \epsilon$ is bounded above by*

$$O\left(\frac{m}{\epsilon}\|\lambda^{(0)} - \lambda^\star\|_2\right) \text{ operations.} \tag{32}$$

4.3 Complexity analysis

At this point, the convergence analysis is not complete since $\|\lambda^{(0)} - \lambda^\star\|_2$ could be arbitrarily large. In order to compare the proposed first order method with the interior point method from [20], we need to upper-bound this quantity. Unfortunately, we did not manage to find a universal bound. Instead, we propose a detailed complexity analysis in the Appendix B. The main conclusions are given in the following theorem.

Theorem 1 *The number of operations needed by Algorithm 1 to reach a relative accuracy of the form:*

$$\frac{\|\alpha^{(k)} - \alpha^\star\|_2}{\|\alpha^{(0)} - \alpha^\star\|_2} \leq \eta \tag{33}$$

varies from $O\left(\frac{m}{\eta}\right)$ operations for simple problems and can reach $O\left(\frac{m^2}{\eta}\right)$ operations for adversarial problems.

Unfortunately the worst-case complexity $O\left(\frac{m^2}{\eta}\right)$ would make the proposed approach irrelevant. In the practical examples that we treated, the ratio $\frac{\|\lambda^\star\|_2}{\|A^T\lambda^\star\|_2}$ however remained bounded by values never exceeding 100, explaining the rather good behavior of the proposed method.

4.3.1 Comparison with existing solvers

Isotonic regression on a DAG received a considerable attention in the optimization literature lately. The recent reference [20] is seemingly the best approach available so far. Therein, the authors propose to use an interior point algorithm [38] exploiting the special graph structure of the matrix A [39]. Their tailored algorithm provides a feasible estimate $\alpha^{(\epsilon)}$ of $\alpha^\star$ satisfying $A\alpha^{(\epsilon)} \geq 0$ with $\|\alpha^{(\epsilon)} - \alpha^\star\|_2^2 \leq \epsilon$ in no more than

$$O(m^{1.5}\log^2 p \log(p/\epsilon)) \tag{34}$$

operations. This worst-case bound is significantly better than ours, both in terms of dimension and precision. The source code is provided here: `https://github.com/sachdevasushant/Isotonic`. Unfortunately in our experiments, the algorithm worked nicely for small m, but systematically failed to converge when dealing with the large graphs appearing in our problems. This problem seems to be related to some instabilities of the current fast randomized Laplacian solvers for large scale graphs [39].

An alternative algorithm was proposed in [18], where the authors proposed to solve a sequence of linear programs `https://www.tau.ac.il/~saharon/files/IRPv1.zip` to reach the solution by partitioning the graph. This algorithm has a worst case complexity of order $O(n^4)$ [5], which is far too large. Fortunately, the worst-case analysis seems to be very pessimistic and the algorithm implemented with MOSEK [40] is much more efficient in practice. From a practical point of view, our experiments on real data showed that: i) the time needed to find exact solutions on real images was significantly too large, but ii) the computing times to get approximate solutions are on par with the ones obtained by our first-order approach for a similar precision with our method becoming preferable with large images. To our belief, the main advantage of our approach is that it is based on simple and portable algorithms, while the recursive partitioning approach in [18] requires the use of heavy large scale linear programming solvers such as MOSEK.

5 Some properties of the models

In this section, we propose to analyze some of the models properties.

5.1 Local mean preservation

An important property of the three models is that they promote piecewise constant images and that the value of the solution on the constant parts are equal to the mean of u_0 over the parts. This is reminiscent of the total variation regularized solutions which suffer from staircasing. An important difference however is the mean preservation: the Rudin-Osher-Fatemi model [41] preserves the mean of the image globally, but the mean on the constant part is not preserved. This produces an undesirable bias in the jump set [42]. The models proposed in this paper do not suffer from this drawback as proved below.

We consider the problem (28) which encompasses (20) and (11) as specific instances. We let $\alpha^\star$ denote its solution.

[5] As far as we could judge, there seems to be an inaccuracy in the complexity analysis, which is based on the exact resolution of linear programs using interior point methods (which are inexact in nature). In practice the implementation is based on a simplex-type algorithm which is exact, but with an uncontrolled complexity.

Theorem 2 (Mean preservation) *Let (B_k) denote the partition of (V,G) into connected components, such that $\alpha^\star|_{B_k}$ is constant and $\alpha^\star|_{B_k} \neq \alpha^\star|_{B_j}$ if B_k is adjacent to B_j. Then*

$$\alpha^\star|_{B_k} = \frac{\sum_{i\in B_k} w_i\beta_i}{\sum_{i\in B_k} w_i}. \tag{35}$$

The proof is reported to the appendix, see Section C. Let us mention that the Theorem 2 is rather standard in the literature of isotonic regressions with a slightly refined notion of partition [17,43]. It shows that solving an isotonic regression is equivalent to partitioning the graph. This is the main idea underlying methods such as [18]. The following result directly follows from Theorem 2.

Corollary 1 (Global mean preservation and maximum principle) *Let $u^\star$ be any of the three solutions $u^\star_{glo}$, $u^\star_{loc1}$ or $u^\star_{loc2}$. Then $\bar{u}^\star(\Omega) = \bar{u}_0(\Omega)$ and*

$$\min_{x\in\Omega} u_0(x) \leq \min_{x\in\Omega} u^\star(x) \leq \max_{x\in\Omega} u^\star(x) \leq \max_{x\in\Omega} u_0(x). \tag{36}$$

5.2 Inclusion of models

The following inclusion relationship holds between the 3 models.

Theorem 3 *For any image u_1, we have*

$$\mathcal{U}_{glo} \subseteq \mathcal{U}_{loc1} \subseteq \mathcal{U}_{loc2}. \tag{37}$$

Hence, $SNR_{glo} \leq SNR_{loc1} \leq SNR_{loc2}$.

The proof is reported to the appendix, see Section D.

5.3 Regularity properties

The functions Δ_{glo}, Δ_{loc1} and Δ_{loc2} can be seen as the Moreau-Yosida regularization of the indicators of the sets $\mathcal{U}_{glo}$, $\mathcal{U}_{loc1}$, $\mathcal{U}_{loc2}$. Hence, the following proposition directly follows.

Proposition 4 (Convexity and regularity) *Let u_1 denote an arbitrary image. Let $\Delta : u \mapsto \Delta(u)$ denote any of the functionals $\Delta_{glo}(u_1,u)$, $\Delta_{loc1}(u_1,u)$ or $\Delta_{loc2}(u_1,u)$ and $u^\star$ denote the associated minimizer. Then:*

- *The function Δ is convex and lower semi-continuous.*
- *The function Δ is differentiable with a 1-Lipschitz continuous gradient.*
- *The gradient of Δ is given by $\nabla\Delta(u) = u - u^\star$.*

The proposition 4 may be useful to design numerical procedures when the metric Δ is used within a variational framework.

5.4 Invariance properties

To finish this theoretical study, let us mention a few properties of the different SNRs introduced in this paper. The notation $S(u_1,u_0)$ stands for any of the measures SNR_{glo}, SNR_{loc1} or SNR_{loc2}.

- The SNRs are invariant to linear and affine transforms of gray levels with a coefficient $a \geq 0$, i.e.

$$S(au_1 + b, u_0) = S(u_1,u_0), \quad \forall a \geq 0. \tag{38}$$

- They are invariant to global contrast changes, by the inclusions of models in Theorem 3. For all non decreasing functions $\phi : \mathbb{R} \to \mathbb{R}$, we get:

$$S(\phi(u_1),u_0) = S(u_1,u_0). \tag{39}$$

  Similarly, SNR_{loc1} and SNR_{loc2} are invariant to local contrast changes on a tree by construction and by Theorem 3, while SNR_{loc2} is invariant to local contrast changes on a DAG by construction.
- The SNRs are not invariant to isometries in the space domain (i.e. translations and rotations of the image) due to discretization issues. However, the continuous counterparts of the measures are invariant.
- For all images u_0 and u_1, we have $S(u_1,0) = \infty$ and $S(0,u_0) = -10\log_{10}(|\Omega|\mathrm{Var}_\Omega(u_0)/\|u_0\|_2^2)$.
- In general, the SNRs are not symmetric:

$$S(u_1,u_0) \neq S(u_0,u_1).$$

  However, it is possible to make them symmetric by computing $\max(S(u_1,u_0), S(u_0,u_1))$.

5.5 Invariance to real illumination changes

At this point the reader might object that transformations preserving the level-lines are far from approaching the complexity of real world illumination changes. This is definitely a valid objection.

The first counter-example that comes to mind is probably that of shadows: they create new level-lines and shapes in the images. Hence they would be detected as changes in our model, while they are simply due to variations of illuminations. Even if shadows are neglected, it is quite easy to see that the most realistic models of image formation based on ray tracing or involving the bidirectional reflectance distribution function (BRDF) would yield significantly more complex phenomena. Even in the case of a Lambertian model of reflectance, it was shown in [5] that only 3D developable surfaces make the level-lines invariant to illumination

change. Overall the proposed model only accounts for very simple models involving occlusions, transparencies and variations of color and reflectance as basic bricks, with no care for the differential geometry of the underlying 3D scene. We refer the interested reader to [3] for a more exhaustive treatment of this model.

To our belief, the simplicity of this model also makes its strength: measuring the similarity of images can be achieved by solving *convex* problems with a structure amenable to efficient numerical algorithms. Using realistic models would most probably lead to hard non-convex problems with no guaranteed complexity.

5.6 Stability issues

The stability of the topographic maps to digitization (blur and sampling) has been studied and proved in [3].

The proposed method is however not stable to noise on the reference image u_1. Indeed, the noise creates spurious level-lines which can be amplified by the projection procedure. This effect is illustrated in Fig. 6. In this experiment, we compare the two images in Fig. 6a and in Fig. 6e. The image in Fig. 6e corresponds to u_1 and contains indiscernible level-sets in the background indicated with a red arrow. The projection procedure yields the image in Fig. 6d. As can be seen there, a significant amplification of the level sets took place, creating an unnatural mark on the image.

Mitigating this effect might be performed by denoising the reference image u_1. The simplest way to do it is probably to quantize the image, which also presents the interest of speeding-up the algorithms by reducing the graphs sizes. A better option is to use more advanced denoisers for the reference image u_1.

In our numerical experiments, we simply used images quantized on 255 levels.

6 Numerical results

6.1 Image comparison and change detection

In order to assess the relevance of the proposed approach for image comparison and change detection, we took pictures of two scenes - denoted F and G - under different lighting conditions (window shutter closed or open). We then evaluated $u^\star_{glo}$, $u^\star_{loc1}$ and $u^\star_{loc2}$ as well as their differences with u_0 for the same scenes under different illuminations, or different scenes under a similar illumination. The results are displayed in Fig. 5 and 6. As can be seen in these experiments, the SNR between the two pairs of images is low (below 12dB), due to the change of illumination and to the change of scene.

As expected from the inclusion of models reported in Theorem 3, we have $SNR \leq SNR_{glo} \leq SNR_{loc1} \leq SNR_{loc2}$. The three algorithms are capable of transforming the gray-levels of u_1 so that they match those of u_0 quite well. An important difference between the models is the area of the constant zones: the image $u^\star_{glo}$ for instance has very large areas with constant gray values while the areas decrease for $u^\star_{loc1}$ and even more for $u^\star_{loc2}$. This is once again an effect reflecting the inclusion of models from Theorem 3.

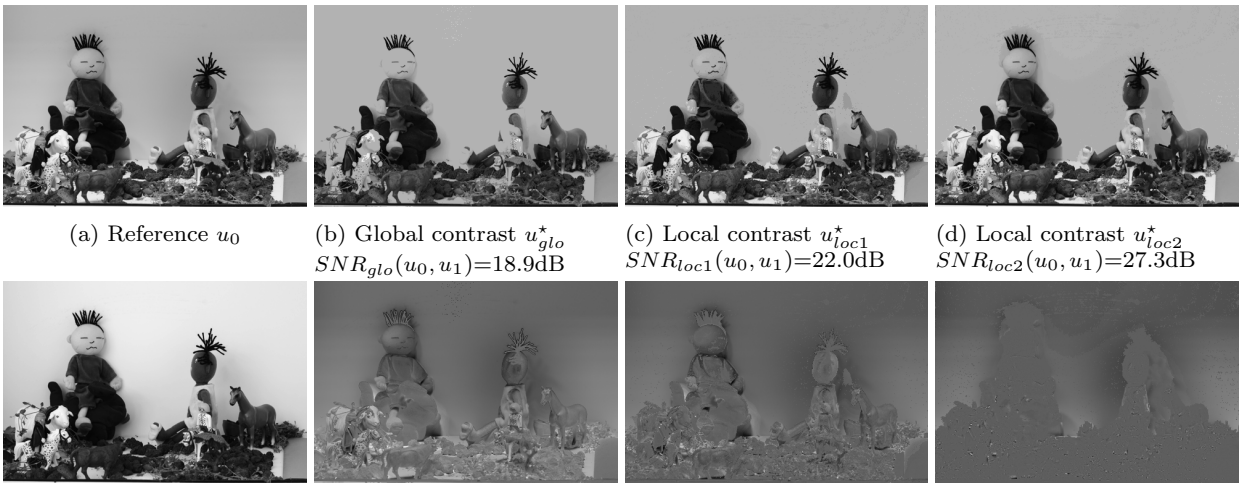
It is quite instructive to look at the difference images, especially for the two different scenes from Fig. 6. What is seen there is that the algorithms tend to outline the objects in the picture u_0 which are not in u_1: from one scene to the next, the black doll, the dinosaur tail and the white doll's head moved and this is exactly what is outlined in the difference, especially in the models $u^\star_{loc2}$ and $u^\star_{loc1}$. In Fig. 6i, 6k and 6j, we also displayed the other possible difference, which consists in projecting u_1 onto a set of images defined through u_0. It is interesting to see that the algorithm outlines the other changes: the objects in u_1 which are not in u_0. This experiment shows the asymmetry of the model enunciated in Section 5.4.

In the differences from Fig. 5, the scenes are identical and the residuals therefore outline the types of illumination changes which are not captured by the proposed models. It is quite clear that some of the specularities and shadows are still visible. For instance, the background in the scence is the base plate of a metallic rack, which is specular and slightly curved (i.e. the perception of gray-values depends on the position of the observer with respect to the light source). This part of the image is partly detected as a difference.

6.2 A large panel of illumination changes

In this section, we evaluate the different signal-to-noise ratio between all pairs of images in Fig. 7. The results are displayed in Tables 2, 3, 4 and 5. As can be seen in these tables, the SNR between images corresponding to identical scenes is higher than that of images corresponding to different scenes, for all notions of SNRs, except the usual one which seems less discriminative.

We can see the different measures of similarity as clustering algorithms: they map pairs of scenes to $\mathbb{R}$. A good algorithm should be able to cluster identical scenes to similar parts of the real line and different scenes to another location. To make this observation quantitative, we can compute the similarity measure for each pair (u_0, u_1). In this section, we will use the



(a) Reference u_0 (b) Global contrast $u^\star_{glo}$ $SNR_{glo}(u_0,u_1)$=18.9dB (c) Local contrast $u^\star_{loc1}$ $SNR_{loc1}(u_0,u_1)$=22.0dB (d) Local contrast $u^\star_{loc2}$ $SNR_{loc2}(u_0,u_1)$=27.3dB

(e) Illumination change u_1 $SNR(u_0,u_1)$=11.4dB (f) Difference $u_0-u^\star_{glo}$ (g) Difference $u_0-u^\star_{loc1}$ (h) Difference $u_0-u^\star_{loc2}$

Fig. 5: Comparing the different algorithms for identical scenes under different illuminations.

symmetric relative mean square error defined by:

$$RMSE(u_1,u_0) := \max\left(\frac{\|u^\star(u_1)-u_0\|_2^2}{\|u_0\|_2^2}, \frac{\|u^\star(u_0)-u_1\|_2^2}{\|u_1\|_2^2}\right),$$

where for $i\in\{0,1\}$

$$u^\star(u_i) = \operatorname*{argmin}_{u\in\mathcal{U}(u_i)} \frac{1}{2}\|u-u_{1-i}\|_2^2$$

and $\mathcal{U}(u_i)$ is any of the sets described previously, defined from an image u_i. We expect the RMSE to be close to 0 when u_1 and u_0 come from the same scene and to be larger when they come from different scenes. Using all the pairs, we can construct two discrete probability distributions $\mathcal{L}_1$ and $\mathcal{L}_2$ corresponding to the RMSE of identical scenes and of different scenes respectively. After computing their means μ_1 and μ_2 and variance σ_1^2 and σ_2^2, we can compare their ability to cluster the pairs of identical and different scenes by computing the ratio

$$z=\frac{|\mu_1-\mu_2|}{\sqrt{\sigma_1^2+\sigma_2^2}}.$$

Assuming that the distributions are Gaussian, this ratio can be interpreted as a z-score which measures how well the two distributions are separated relative to their standard deviation. The higher the ratio, the better. For instance, a ratio of order 3 indicates that a test can be built that will fail for less than 3 out of 1000 pairs of images. With a ratio of 2, the test will fail about 5 percent of the time. Table 1 shows the different results. As can be seen from this table and for the set of images used to construct the test, the best algorithm is SNR_{loc1} followed by SNR_{loc2}, SNR_{glo}, SNR and $SSIM$. Those results are very preliminary, subject to statistical issues and more extensive tests should be pursued. Still, this preliminary comparison shows that the proposed measures of similarity are quite promising.

$RMSE$	$SSIM$	$RMSE_{glo}$	$RMSE_{loc1}$	$RMSE_{loc2}$
1.86	1.63	2.89	3.27	3.00

Table 1: z-scores of the different measures to discriminate whether pairs of images come from similar or different scenes under different illuminations.

Acknowledgment

The authors wish to thank the anonymous reviewers for their excellent reviews which helped them improving the paper. They thank Pascal Monasse for encouraging them to explore this question and for providing the source codes of the FLST (Fast Level Set Transform). P. Weiss wishes to thank Jonas Kahn warmly for helping him to check that dynamic programming could be applied to our problem and to find a hard problem. He also thanks his daughter Anouk for lending her toys to generate the pictures. P. Escande was supported by the PRES of Toulouse University and Midi-Pyrénées region.

	F_1	F_2	F_3	F_4	F_5	G_1	G_2	G_3	G_4	G_5
F_1	Inf	12.70	10.88	10.84	16.81	9.97	9.90	9.91	8.61	8.69
F_2	11.26	Inf	19.24	14.72	9.91	8.07	8.61	8.44	9.99	8.73
F_3	9.69	19.49	Inf	12.44	8.81	7.23	7.85	7.72	9.35	7.72
F_4	8.59	13.92	11.38	Inf	8.79	8.52	8.88	8.56	10.81	10.68
F_5	16.96	11.50	10.16	11.19	Inf	10.56	10.46	10.52	8.86	8.82
G_1	9.18	8.72	7.63	9.97	9.62	Inf	24.55	23.99	13.54	14.32
G_2	9.05	9.20	8.19	10.28	9.46	24.49	Inf	32.08	15.35	14.36
G_3	9.22	9.19	8.22	10.12	9.68	24.09	32.24	Inf	15.04	13.84
G_4	6.80	9.62	8.73	11.24	6.90	12.52	14.39	13.92	Inf	14.20
G_5	6.51	7.99	6.74	10.76	6.50	12.94	13.04	12.36	13.84	Inf

Table 2: Standard SNRs between all pairs.

	F_1	F_2	F_3	F_4	F_5	G_1	G_2	G_3	G_4	G_5
F_1	Inf	18.11	14.60	16.05	17.24	11.65	11.21	11.14	10.26	12.47
F_2	18.88	Inf	23.91	18.47	15.52	11.51	11.32	11.26	11.00	12.79
F_3	17.05	25.54	Inf	18.45	14.74	11.56	11.40	11.33	11.12	12.87
F_4	14.94	17.38	16.83	Inf	18.04	11.98	11.81	11.75	11.60	13.17
F_5	17.14	15.01	13.03	18.82	Inf	12.24	11.75	11.69	10.64	12.79
G_1	10.54	9.49	8.57	11.44	11.21	Inf	29.56	29.20	18.08	20.20
G_2	10.58	9.68	8.83	11.59	11.24	29.34	Inf	36.34	19.61	21.07
G_3	10.58	9.68	8.82	11.60	11.24	28.81	36.28	Inf	19.76	21.12
G_4	10.41	10.34	9.71	12.46	10.97	19.51	21.59	21.63	Inf	21.49
G_5	10.06	9.48	8.69	11.42	10.58	18.28	19.30	19.30	19.03	Inf

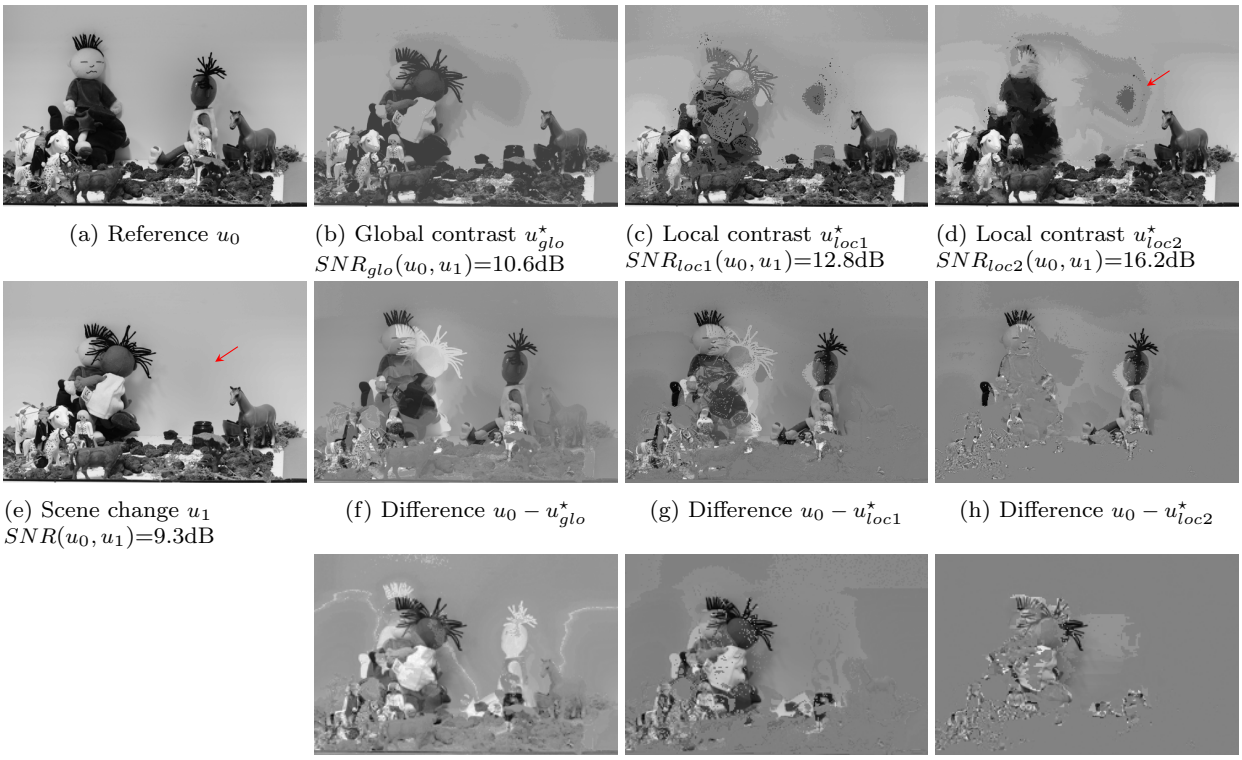
Table 3: SNR_{glo} between all pairs

	F_1	F_2	F_3	F_4	F_5	G_1	G_2	G_3	G_4	G_5
F_1	Inf	20.72	17.27	18.39	19.36	13.56	13.19	13.13	12.38	14.19
F_2	22.00	Inf	26.06	20.38	17.45	13.16	12.99	12.95	12.85	14.33
F_3	19.94	27.93	Inf	20.09	16.62	13.07	12.94	12.89	12.82	14.32
F_4	17.37	19.85	19.22	Inf	21.11	13.78	13.58	13.54	13.49	14.90
F_5	19.83	17.99	15.99	21.49	Inf	14.29	13.83	13.77	12.86	14.59
G_1	12.80	11.86	10.90	13.89	13.53	Inf	31.99	31.04	20.30	22.20
G_2	12.79	12.01	11.14	13.92	13.53	32.32	Inf	37.78	21.39	23.33
G_3	12.75	12.07	11.24	14.01	13.49	31.17	37.65	Inf	21.90	23.45
G_4	12.03	11.92	11.19	14.15	12.68	22.42	24.39	24.61	Inf	25.23
G_5	11.80	11.37	10.54	13.47	12.37	20.52	21.69	21.73	21.84	Inf

Table 4: SNR_{loc1} between all pairs

	F_1	F_2	F_3	F_4	F_5	G_1	G_2	G_3	G_4	G_5
F_1	Inf	26.03	21.69	23.50	24.64	17.06	16.62	16.53	15.67	17.51
F_2	27.67	Inf	38.62	26.15	22.82	17.15	17.08	17.05	17.01	18.11
F_3	27.07	43.07	Inf	25.77	22.27	17.02	16.94	16.90	16.83	18.04
F_4	22.80	25.91	26.25	Inf	27.45	18.11	17.99	18.00	17.73	18.84
F_5	25.06	23.24	20.64	26.08	Inf	17.65	17.15	17.07	16.08	17.96
G_1	16.49	15.02	13.71	17.04	16.95	Inf	41.90	39.20	24.91	29.21
G_2	16.83	15.79	14.57	17.46	17.42	41.11	Inf	44.36	24.90	30.40
G_3	16.47	15.75	14.65	17.52	16.99	38.70	47.12	Inf	26.51	31.18
G_4	15.44	15.41	14.68	17.63	15.92	28.62	32.78	33.05	Inf	33.71
G_5	15.16	14.49	13.36	16.71	15.55	27.52	30.72	30.77	28.68	Inf

Table 5: SNR_{loc2} between all pairs



(a) Reference u_0
(b) Global contrast $u^\star_{glo}$ $SNR_{glo}(u_0,u_1)$=10.6dB
(c) Local contrast $u^\star_{loc1}$ $SNR_{loc1}(u_0,u_1)$=12.8dB
(d) Local contrast $u^\star_{loc2}$ $SNR_{loc2}(u_0,u_1)$=16.2dB
(e) Scene change u_1 $SNR(u_0,u_1)$=9.3dB
(f) Difference $u_0-u^\star_{glo}$
(g) Difference $u_0-u^\star_{loc1}$
(h) Difference $u_0-u^\star_{loc2}$
(i) Other global difference
(j) Other local 1 difference
(k) Other local 2 difference

Fig. 6: Comparing the different algorithms for different scenes under a similar illumination.

(a) F_1 (b) F_2 (c) F_3 (d) F_4 (e) F_5
(f) G_1 (g) G_2 (h) G_3 (i) G_4 (j) G_5

Fig. 7: Different images used for comparison

A Proofs of convergence of the first order algorithm

We first prove Proposition 1.

Proof We only sketch the proof. The idea is to use Fenchel-Rockafellar duality for convex optimization:

$$\begin{aligned}&\min_{A\alpha\geq 0}\frac{1}{2}\langle W(\alpha-\beta),\alpha-\beta\rangle\\&=\min_{\alpha\in\mathbb{R}^m}\sup_{\lambda\leq 0}\frac{1}{2}\langle W(\alpha-\beta),\alpha-\beta\rangle+\langle A\alpha,\lambda\rangle\\&=\sup_{\lambda\leq 0}\min_{\alpha\in\mathbb{R}^m}\frac{1}{2}\langle W(\alpha-\beta),\alpha-\beta\rangle+\langle A\alpha,\lambda\rangle.\end{aligned}$$

The primal-dual relationship $\alpha(\lambda)$ is obtained by finding the minimizer of the inner-problem in the last equation. The dual problem is found by replacing α by $\alpha(\lambda)$ in the inner-problem.

The function D is obviously differentiable with $\nabla D(\lambda) = -AW^{-1}A^T\lambda + A\beta$. Therefore, $\forall(\lambda_1, \lambda_2)$, we get:

$$\begin{aligned}\|\nabla D(\lambda_1) - \nabla D(\lambda_2)\|_2 &= \|AW^{-1}A^T(\lambda_1 - \lambda_2)\|_2 \\ &\leq \lambda_{\max}(AW^{-1}A^T)\|\lambda_1 - \lambda_2\|_2.\end{aligned}$$

The inequality (30) is a direct consequence of a little known result about the Fenchel-Rockafellar dual of problems involving a strongly convex function. We refer the reader to Lemma D.1 in [44] for more details, or to [7] for a slightly improved bound in the case of ℓ^2 metrics.

Now let us prove Proposition 2.

Proof Notice that $\lambda_{\max}(AW^{-1}A^T) = \sigma^2_{\max}(AW^{-1/2})$, where $\sigma_{\max}$ stands for the largest singular value. Moreover

$$\begin{aligned}\|AW^{-1/2}\alpha\|_2^2 &= \sum_{k=1}^{m}\left(\frac{\alpha_{\mathrm{I}(k)}}{\sqrt{w_{\mathrm{I}(k)}}} - \frac{\alpha_{\mathrm{J}(k)}}{\sqrt{w_{\mathrm{J}(k)}}}\right)^2 \\ &\leq \sum_{k=1}^{m} 2\left(\frac{\alpha^2_{\mathrm{I}(k)}}{w_{\mathrm{I}(k)}} + \frac{\alpha^2_{\mathrm{J}(k)}}{w_{\mathrm{J}(k)}}\right) \\ &= 4\sum_{k=1}^{m}\frac{\alpha^2_{\mathrm{I}(k)}}{w_{\mathrm{I}(k)}} \\ &= 4\sum_{i=1}^{p} n_i\frac{\alpha_i^2}{w_i},\end{aligned}$$

where n_i denotes the number of edges starting from vertex i (the outdegree). To conclude, notice that each pixel in region Δ_j has at most $c_{\max}$ neighbors. Therefore $n_i \leq w_i c_{\max}$ and we finally get:

$$\|AW^{-1/2}\alpha\|_2^2 \leq 4c_{\max}\sum_{i=1}^{p}\alpha_i^2 \qquad = 4c_{\max}\|\alpha\|_2^2. \tag{40}$$

Finally, we prove Proposition 3 below.

Proof Standard convergence results [37] state that:

$$D(\lambda^{(k)}) - D(\lambda^\star) \leq \frac{4c_{\max}\|\lambda^{(0)} - \lambda^*\|_2^2}{k^2}.$$

Combining this result with inequality (30) directly yields (31).

To obtain the bound (32), first remark that each iteration of Algorithm 1 requires two matrix-vector products with A and A^T of complexity $O(m)$. The final result is then a direct consequence of the bound (31) and of the Proposition 2.

B Proofs of the complexity results

In this paragraph, we analyze the theoretical efficiency of Algorithm 1. We consider the special case $W = \mathrm{Id}$ for the ease of exposition. In practice, controlling the *absolute error* $\|\alpha^{(k)} - \alpha^\star\|_2$ is probably less relevant than the *relative error* $\frac{\|\alpha^{(k)} - \alpha^\star\|_2}{\|\alpha^{(0)} - \alpha^\star\|_2}$. This motivates setting $\epsilon = \eta\|\alpha^{(0)} - \alpha^\star\|_2$ in equation (32), where $\eta \in [0, 1)$ is a parameter describing the relative precision of the solution. Setting $\lambda^{(0)} = 0$ and noticing that:

$$\begin{aligned}\|\alpha^{(0)} - \alpha^\star\|_2 &= \|\beta - \alpha^\star\|_2 \\ &= \|A^T\lambda^\star\|_2,\end{aligned}$$

the complexity in terms of η becomes:

$$O\left(\frac{m}{\eta}\frac{\|\lambda^\star\|_2}{\|A^T\lambda^\star\|_2}\right). \tag{41}$$

Example of a hard problem An example of a hard graph (a simple line graph) is provided in Figure 8. For this graph, the Algorithm 1 can be interpreted as a *diffusion process*, which is known to be extremely slow. In particular, Nesterov shows that diffusions are the worst case problems for the first order methods in [37, p.59].

Proposition 5 *Consider a simple line graph as depicted in Figure 8, with p even and $W = \mathrm{Id}$. Set*

$$\beta_i = \begin{cases} 1 & \text{if } i \leq p/2, \\ -1 & \text{otherwise.}\end{cases} \tag{42}$$

Then the primal-dual solution $(\alpha^\star, \lambda^\star)$ of the isotonic regression problem (28) is given by $\alpha^\star = 0$ and

$$\lambda_k^\star = \begin{cases} -k & \text{if } \ 1 \leq k \leq p/2, \\ -n + k & \text{if } \ p/2 + 1 \leq k \leq p.\end{cases} \tag{43}$$

This implies that

$$\frac{\|\lambda^\star\|_2}{\|A^T\lambda^\star\|_2} \sim m. \tag{44}$$

Proof For this simple graph, $m = p - 1$. To check that (43) is a solution, it suffices to verify the Karush-Kuhn-Tucker conditions:

$$\begin{aligned} &A^T\lambda^\star = W(\beta - \alpha^\star), \\ &A\alpha^\star \geq 0, \\ &\lambda^\star \leq 0, \\ &\lambda_i^\star = 0 \qquad\qquad\qquad\qquad \text{if } (A\alpha^\star)_i > 0.\end{aligned}$$

This is done by direct inspection, using the fact that for this graph:

$$(A^T\lambda)_i = \begin{cases} -\lambda_1 & \text{if} \quad i = 1 \\ -\lambda_i + \lambda_{i-1} & \text{if} \quad 2 \leq i \leq p-1 \\ \lambda_{p-1} & \text{if} \quad i = p.\end{cases} \tag{45}$$

The relationship (44) is due to the fact that the sum of squares $\sum_{k=1}^m k^2 = m(m+1)(2m+1)/6 \sim m^3$ so that $\|\lambda^\star\|_2^2 \sim m^3$ and $\|A^T\lambda^\star\|_2^2 = \|\beta\|_2^2 = m$.

Fig. 8: Worst case graph

Example of a nice problem In order to rehabilitate our approach, let us show that the ratio $\frac{\|\lambda^\star\|_2}{\|A^T\lambda^\star\|_2}$ can be bounded independently of m for "nice" graphs.

Proposition 6 *For any $\lambda \leq 0$ and for the graph depicted in Figure 10, we have:*

$$\frac{1}{2} \leq \frac{\|\lambda^\star\|_2}{\|A^T\lambda^\star\|_2} \leq \frac{1}{\sqrt{2}}. \tag{46}$$

Proof For this graph, we get:

$$(A^T\lambda)_i = \begin{pmatrix} -\lambda_1 \\ \lambda_1 + \lambda_2 \\ -\lambda_2 - \lambda_3 \\ \vdots \\ \lambda_{n-2} + \lambda_{n-1} \\ -\lambda_{n-1}\end{pmatrix}. \tag{47}$$

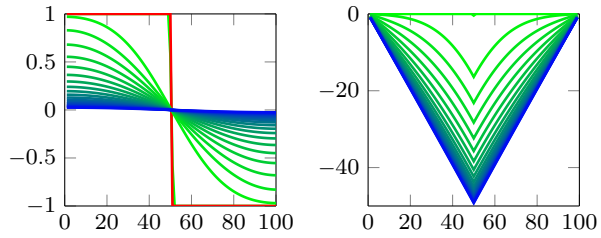


Fig. 9: First 20000 iterations of the primal-dual pair $(\alpha^{(k)}, \lambda^{(k)})$. Top: β is displayed in red while $\alpha^{(k)}$ varies from green to blue with iterations. Bottom: $\lambda^{(k)}$ varies from green to blue with iterations. A new curve is displayed every 1000 iterations. As can be seen, the convergence is very slow.

Therefore:

$$\begin{aligned}\|A^T\lambda\|_2^2 &= \lambda_1^2 + \lambda_{n-1}^2 + \sum_{k=1}^{n-2}(\lambda_k + \lambda_{k+1})^2 \\ &= 2\sum_{k=1}^{n-1}\lambda_k^2 + 2\sum_{k=1}^{n-2}\lambda_k\lambda_{k+1},\end{aligned}$$

and

$$2\|\lambda\|_2^2 \leq \|A^T\lambda\|_2^2 \leq 4\|\lambda\|_2^2. \tag{48}$$

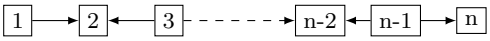


Fig. 10: A nice graph

C Proof of local mean preservation

We prove Theorem 2 below.

Proof The Karush-Kuhn-Tucker optimality conditions read:

$$w_i(\alpha_i^\star - \beta_i) + (A^T\lambda)_i = 0 \qquad \forall i \tag{49}$$

$$A\alpha \geq 0 \tag{50}$$

$$\lambda \leq 0 \tag{51}$$

$$\lambda_{i,j}(A\alpha^\star)_{i,j} = 0 \qquad \forall (i,j) \in E, \tag{52}$$

with $(A^T\lambda)_i = \sum_{j,(i,j)\in E}\lambda_{i,j} - \sum_{j,(j,i)\in E}\lambda_{i,j}$. Hence we get:

$$\sum_{i\in B_k}(A^T\lambda)_i = \sum_{i\in B_k}\sum_{j,(i,j)\in E}\lambda_{i,j} - \sum_{j,(j,i)\in E}\lambda_{i,j} = 0.$$

To obtain the last equality, observe that the Lagrange multipliers $\lambda_{i,j}$ can be separated into those joining B_k from the exterior and those linking two edges within B_k. The first ones vanish thanks to (52) and to the assumption that $(A\alpha^\star)_{i,j} \neq 0$. The other ones cancel since the neighborhood is symmetric. To conclude the proof, it suffices to sum the equation (49) over B_k.

D Proof of the models inclusion

To prove Theorem 3, we first need the following preparatory lemma.

Lemma 2 (Directed paths in the tree of shapes) *Let u_1 denote an image and (V, E) denote the graph associated to its tree of shapes $(\omega_i)_{i\in I}$. Let x and y be adjacent points $x \overset{\mathcal{N}_4}{\sim} y$ and ω_i (resp. ω_j) denote the smallest shape containing x (resp. y).*

Then there exists a path $(i_0 = i, i_1, \dots, i_{l-1}, i_l = j)$ in E linking ω_i to ω_j. In addition $\operatorname{sign}(s_{i_k}) = \operatorname{sign}(u_1(y) - u_1(x))$ for $1 \leq k \leq l$.

Proof We have $x \in \partial\omega_i$ since ω_i is the smallest containing x. Otherwise, there would exist a descendant (which would be smaller by definition of the tree) that contains x. Similarly, $y \in \partial\omega_j$.

Second, we have $\omega_i \subset \omega_j$ or $\omega_j \subset \omega_i$, but $\omega_i \cap \omega_j = \emptyset$ is not possible. If it were the case, then ω_i and ω_j would be shapes on different branches of the tree. This is impossible since elements on different branches are disconnected.

Note that $u_1(x) \neq u_1(y)$, otherwise they would be in the same shape. In what follows, we assume that $\omega_i \subset \omega_j$ and that $u_1(y) > u_1(x)$. The 3 other cases can be treated similarly. We let $(i_0 = i, i_1, \dots, i_{l-1}, i_l = j)$ denote the path in E linking ω_i to ω_j. We claim that along this path $(s_{i_k})_{1\leq k\leq l}$ is constant and equal to 1. There is necessarily one sign $s_{i_k} = 1$, otherwise this would contradict the hypothesis $u_1(y) > u_1(x)$, so that the result holds when $l = 1$. When $l > 1$, let us assume that there exists one sign equal to -1. Then, there exists two consecutive indexes, say i_{k_0} and i_{k_0+1}, with $1 \leq k_0 \leq l$ such that $s_{i_{k_0}} = -1$ and $s_{i_{k_0+1}} = 1$ (or the reverse). This implies that $\omega_{i_{k_0}}$ is a shape from the min-tree and that $\omega_{i_{k_0+1}}$ is a shape from the max-tree (see [22] or the introduction of [30]). Therefore, $\omega_{i_{k_0+1}}$ is a cavity of $\omega_{i_{k_0}}$, so that ω_j is not adjacent to ω_i, contradicting $x \overset{\mathcal{N}_4}{\sim} y$.

We are now ready to prove Theorem 3.

Proof We first prove the inclusion $\mathcal{U}_{glo} \subseteq \mathcal{U}_{loc1}$. Assume that $u \in \mathcal{U}_{glo}$. Then for all $(x, y) \in \Omega^2$, $(u(x) - u(y)) \cdot (u_1(x) - u_1(y)) \geq 0$. Therefore, the constraint $A\alpha \geq 0$ is verified since it describes differences of gray values in adjacent level-lines. In addition $u_1(x) = u_1(y) \Rightarrow u(x) = u(y)$. Hence, the constant regions of u_1 are preserved.

We now prove the inclusion $\mathcal{U}_{loc1} \subseteq \mathcal{U}_{loc2}$. The property $[x \overset{\mathcal{N}_4}{\sim} y \text{ and } u_1(x) = u_1(y)] \Rightarrow [u(x) = u(y)]$ is obvious since it implies that x and y belong to the level line $\partial\omega_i$. Let $u \in \mathcal{U}_{loc1} = \{R\alpha, A\alpha \geq 0\} = \{L\gamma, \gamma \geq 0\}$. By Lemma 2, we deduce that $x \overset{\mathcal{N}_4}{\sim} y$ implies that $(u(x) - u(y))(u_1(x) - u_1(y)) \geq 0$. For instance assume that $u_1(y) > u_1(x)$. Then $u(y) = u(x) + \sum_{1\leq k\leq l}\gamma_{i_k}$ with $\gamma_{i_k} \geq 0$, so that $u(y) - u(x) \geq 0$.

To finish, note that if u_1 is a very simple image such as the one depicted in Fig. 11, $\mathcal{U}_{glo} = \mathcal{U}_{loc1} = \mathcal{U}_{loc2}$, explaining why the inclusion of sets is not strict in general.

References

1. J. Serra, *Image analysis and mathematical morphology*. Academic press, 1982.
2. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *Image Processing, IEEE Transactions on*, vol. 13, no. 4, pp. 600–612, 2004.

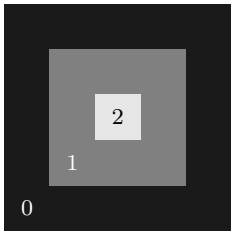


Fig. 11: An example of image u_1 where the three models are equivalent.

3. V. Caselles, B. Coll, and J.-M. Morel, "Topographic maps and local contrast changes in natural images," *International Journal of Computer Vision*, vol. 33, no. 1, pp. 5–27, 1999.
4. C. Ballester, E. Cubero-Castan, M. Gonzalez, and J. Morel, "Contrast invariant image intersection," *Advanced Mathematical Methods in Measurement and Instrumentation*, pp. 41–55, 2000.
5. P. Weiss, A. Fournier, L. Blanc-Féraud, and G. Aubert, "On the illumination invariance of the level lines under directed light: Application to change detection," *SIAM Journal on Imaging Sciences*, vol. 4, no. 1, pp. 448–471, 2011.
6. P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-point algorithms for inverse problems in science and engineering*. Springer, 2011, pp. 185–212.
7. A. Chambolle and T. Pock, "An introduction to continuous optimization for imaging," *Acta Numerica*, vol. 25, pp. 161–319, 2016.
8. C. Ballester, V. Caselles, L. Igual, J. Verdera, and B. Rougé, "A variational model for P+XS image fusion," *International Journal of Computer Vision*, vol. 69, no. 1, pp. 43–58, 2006.
9. V. Caselles, B. Coll, and J.-M. Morel, "Geometry and color in natural images," *Journal of Mathematical Imaging and Vision*, vol. 16, no. 2, pp. 89–105, 2002.
10. M. J. Ehrhardt and S. R. Arridge, "Vector-valued image processing by parallel level sets," *Image Processing, IEEE Transactions on*, vol. 23, no. 1, pp. 9–18, 2014.
11. M. Droske and M. Rumpf, "A variational approach to nonrigid morphological image registration," *SIAM Journal on Applied Mathematics*, vol. 64, no. 2, pp. 668–687, 2004.
12. A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of mathematical imaging and vision*, vol. 40, no. 1, pp. 120–145, 2011.
13. C. Van Eeden, "Maximum likelihood estimation of partially or completely ordered parameters. i," *Indag. Math*, vol. 19, pp. 128–136, 1957.
14. R. Barlow and H. Brunk, "The isotonic regression problem and its dual," *Journal of the American Statistical Association*, vol. 67, no. 337, pp. 140–147, 1972.
15. R. L. Dykstra, T. Robertson *et al.*, "An algorithm for isotonic regression for two or more independent variables," *The Annals of Statistics*, vol. 10, no. 3, pp. 708–716, 1982.
16. M. J. Best and N. Chakravarti, "Active set algorithms for isotonic regression; a unifying framework," *Mathematical Programming*, vol. 47, no. 1-3, pp. 425–439, 1990.
17. P. M. Pardalos and G. Xue, "Algorithms for a class of isotonic regression problems," *Algorithmica*, vol. 23, no. 3, pp. 211–222, 1999.
18. R. Luss, S. Rosset, and M. Shahar, "Decomposing isotonic regression for efficiently solving large problems," in *Advances in Neural Information Processing Systems*, 2010, pp. 1513–1521.
19. Q. F. Stout, "Isotonic regression via partitioning," *Algorithmica*, vol. 66, no. 1, pp. 93–112, 2013.
20. R. Kyng, A. Rao, and S. Sachdeva, "Fast, Provable Algorithms for Isotonic Regression in all L_p-norms," in *Advances in Neural Information Processing Systems*, 2015, pp. 2701–2709.
21. V. Kolmogorov, T. Pock, and M. Rolinek, "Total variation on a tree," *SIAM Journal on Imaging Sciences*, vol. 9, no. 2, pp. 605–636, 2016.
22. P. Monasse and F. Guichard, "Fast computation of a contrast-invariant image representation," *IEEE Transactions on Image Processing*, vol. 9, no. 5, pp. 860–872, 2000.
23. J. Delon, "Midway image equalization," *Journal of Mathematical Imaging and Vision*, vol. 21, no. 2, pp. 119–134, 2004.
24. A. C. Bovik, *Handbook of image and video processing*. Academic press, 2010.
25. P. Viola and W. M. Wells III, "Alignment by maximization of mutual information," *International journal of computer vision*, vol. 24, no. 2, pp. 137–154, 1997.
26. F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, "Multimodality image registration by maximization of mutual information," *IEEE transactions on Medical Imaging*, vol. 16, no. 2, pp. 187–198, 1997.
27. L. Moisan, *Modeling and Image Processing*. Lectures notes of ENS Cachan, 2012.
28. G. Matheron, *Random sets and integral geometry*. John Wiley & Sons, 1975.
29. V. Caselles and P. Monasse, *Geometric description of images as topographic maps*. Springer, 2009.
30. T. Géraud, E. Carlinet, S. Crozet, and L. Najman, "A quasi-linear algorithm to compute the tree of shapes of nd images," in *International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing*. Springer, 2013, pp. 98–110.
31. A. S. Kronrod, "On functions of two variables," *Uspekhi matematicheskikh nauk*, vol. 5, no. 1, pp. 24–134, 1950.
32. P. F. Felzenszwalb and R. Zabih, "Dynamic programming and graph algorithms in computer vision," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 4, pp. 721–740, 2011.
33. E. Horowitz *et al.*, *Fundamentals of data structures in C++*. Galgotia Publications, 2006.
34. Q. F. Stout, "Fastest isotonic regression algorithms," 2014. [Online]. Available: web.eecs.umich.edu/~qstout/IsoRegAlg_140812.pdf
35. Y. Nesterov, "A method of solving a convex programming problem with convergence rate $o(1/k^2)$," in *Soviet Mathematics Doklady*, vol. 27, no. 2, 1983, pp. 372–376.
36. P. Weiss, L. Blanc-Féraud, and G. Aubert, "Efficient schemes for total variation minimization under constraints in image processing," *SIAM journal on Scientific Computing*, vol. 31, no. 3, pp. 2047–2080, 2009.
37. Y. Nesterov, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2013, vol. 87.
38. Y. Nesterov, A. Nemirovskii, and Y. Ye, *Interior-point polynomial algorithms in convex programming*. SIAM, 1994, vol. 13.
39. D. A. Spielman and S.-H. Teng, "Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems," in *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*. ACM, 2004, pp. 81–90.

40. M. ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 8.1.*, 2017. [Online]. Available: http://docs.mosek.com/8.1/toolbox/index.html
41. L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.
42. C.-A. Deledalle, N. Papadakis, J. Salmon, and S. Vaiter, "Clear: Covariant least-square refitting with applications to image restoration," *SIAM Journal on Imaging Sciences*, vol. 10, no. 1, pp. 243–284, 2017.
43. H. D. Brunk, "Maximum likelihood estimates of monotone parameters," *The Annals of Mathematical Statistics*, pp. 607–616, 1955.
44. C. Boyer, P. Weiss, and J. Bigot, "An algorithm for variable density sampling with block-constrained acquisition," *SIAM Journal on Imaging Sciences*, vol. 7, no. 2, pp. 1080–1107, 2014.