Parameter Tuning by Pairwise Preferences

Pavel Kisilev
pavel.kisilev@hp.com

Daniel Freedman
daniel.freedman@hp.com

Hewlett-Packard Laboratories
Haifa, Israel

Most computer vision algorithms have parameters. This is a fact of life which is familiar to any researcher in the field. Unfortunately, for algorithms to work properly, the parameters have to be tuned. We propose a semi-automatic approach to parameter tuning, which is general-purpose and can be used for a wide variety of computer vision algorithms.

The basic setup is as follows. The vision algorithm takes as input (i) an actual input (commonly an image) and (ii) parameter values. From the input and the parameter values, it produces an output (sometimes an image, sometimes another quantity). Thus, a single run of the vision algorithm may be characterized by the triple $(input, parameter, output)$. The vision algorithm is run several times, leading to several such triples. The user is then given pairs of outputs, and asked to judge which output is preferred, in that it constitutes a higher quality output. The user provides such a pairwise preference for several pairs of outputs. Based on these pairwise preferences, the goal is to find a function over the parameter space which respects the user's preferences, i.e. such that if $parameter_1 \succ parameter_2$ then the function is larger for $parameter_1$ than for $parameter_2$.

Let the parameters be denoted by x. Then this problem can be posed as the following optimization over functions f:

$$\min_{f \in \mathscr{F}} S[f] \quad \text{subject to} \quad f(x_{2i-1}) \geq f(x_{2i}) + 1, \; i = 1, \dots, n$$

where $S[f]$ is a smoothness energy, and the constraints capture the user's pairwise preferences. This is an infinite-dimensional optimization over a continuous function f, but it can be reduced to a convex, finite-dimensional optimization for a particular class of smoothness energy functionals using the theory of Reproducing Kernel Hilbert Spaces (RKHS) [1, 2]. If $S[\cdot]$ can be written as a norm in an RKHS, then f admits a representation of the form

$$f(x) = \sum_{i=1}^{2n} \alpha_i k(x, x_i)$$

where k is the kernel of the RKHS. Given this, the optimization may be written as a quadratic program in the vector α:

$$\min_{\alpha \in \mathbb{R}^{2n}} \alpha^T K \alpha \quad \text{subject to} \quad B\alpha \geq e \tag{1}$$

where K is a $2n \times 2n$ matrix whose elements are $k(x_i, x_j)$; $B = K_{odd} - K_{even}$, for K_{odd} (K_{even}) the $n \times 2n$ submatrix of K with the odd (even) rows selected; and e is the n-dimensional vectors whose entries are all 1's.

Given this preference function $f^*(x)$, we may now find the optimal parameters by maximizing f^*, i.e.

$$x^* = \arg\max_{x \in \mathscr{X}} f^*(x)$$

This latter optimization can be performed exhaustively if the dimension of $\mathscr{X}$ is sufficiently small, i.e. 2 or 3, as is the case in many applications. Alternatively, if the dimension of the parameter space is high, we may approximate the optimal parameter by the best of the parameter values that have been used for learning, i.e.

$$\tilde{x} = \arg\max_{x \in \{x_i\}_{i=1}^{2n}} f^*(x)$$

Experiments on simulated and real world data show the effectiveness of the proposed method. The first set of experiments involve simulated data: we generate a true preference function $f_{true}(x)$. The algorithm obtains the pairs of the form (x_a, x_b) where $f_{true}(x_a) > f_{true}(x_b)$; the function f_{true} values for the pairs (x_a, x_b) are not given to the algorithm, but only their order (i.e. x_a is preferred to x_b). Thus, we effectively simulate the problem of parameter tuning, but we have a ground truth preference function f_{true} we can compare to our approximated preference function.

In the case of 2D parameters, we simulated a 2D preference function f_{true} using a mixture of four 2D Gaussians (Figure 1, left). We generate pairs (x_a, x_b) randomly. Figure 1, right shows the reconstructed function with 16 pairs; we are able to predict correctly the position of maxima, and the general shape of the function.

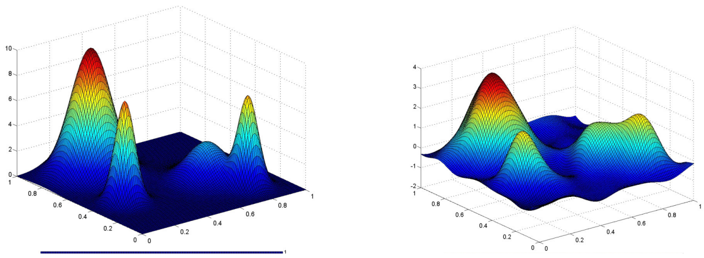

Figure 1: Simulated data examples of "blind" reconstruction of functions. Left: original 2D function; Right: reconstructed 2D function.

Further, we tested our method on image denoising applications. Our image denoiser was a Bilateral Filter (BF) [3] with two parameters, the range bandwidth and the spatial bandwidth; our goal was to determine their optimal values using our algorithm. We added Gaussian noise to images and ran the BF with 16 different combinations of bandwidth parameters; each image received its own setting (4 different settings for each bandwidth parameter). Then, 8 pairs of *different* images were randomly generated and a user judged each pair, selecting the output with better denoising. From these pairwise preferences, the optimal bandwidths were estimated.

To quantify the effectiveness of our algorithm, the following procedure was used. We evaluated the method on a separate "test set," consisting of 30 new images. For the optimal bandwidth parameter setting and the above 16 settings, we computed the averages (over the 30 denoised images in the test set) of two well known image quality measures, the mean squared error (MSE) and the structural similarity measure (SSIM), using our knowledge of the noise-free images. Our method yields the best SSIM index, and is quite close to the best MSE result (see Figure 2). This is a remarkable result as the SSIM index is known to reflect well the visual similarities of images, and our input data are based completely on a visual pairwise comparison test; this further proves the effectiveness of our method.

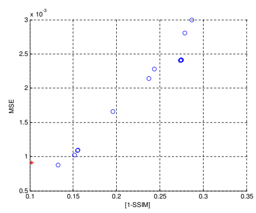

Figure 2: Scatter plot of MSE vs. 1-SSIM of the denoised images; the red star corresponds to the denoising with the estimated optimal parameters.

[1] G. Kimeldorf and G. Wahba. Some results on Tchebycheffian spline functions. *Journal of Mathematical Analysis and Applications*, 33 (1):82–95, 1971.

[2] B. Scholkopf, H. Herbrich, and A.J. Smola. A Generalized Representer Theorem. In *14th Annual Conference on Computational Learning Theory, COLT 2001*, pages 416–426. Springer, 2001.

[3] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on*, pages 839–846, 1998.