Table 4. Barcodes denoising results obtained via PN, Coordinate Optimization and adapted PDHG for the anisotropic model, genuine PDHG for the isotropic model, and a median filter. ISNR (dB) values (higher is better) and running times in seconds are shown.

	ANISOTROPIC				ISOTROPIC		MEDIAN	
SIZE	ISNR	TIME PN	TIME COORD.	TIME PDHG	ISNR	TIME PDHG	ISNR	TIME
100×100	2.39	0.11	2.85	0.64	2.04	0.03	1.24	0.00
175×175	4.14	0.27	15.99	8.71	3.38	0.11	1.74	0.02
300×300	5.48	0.88	140.78	128.72	4.38	0.37	2.35	0.03
375×375	6.04	1.39	167.68	93.87	4.39	0.76	2.42	0.07
500×500	4.42	2.59	228.55	203.19	3.58	1.30	2.18	0.09

(a) (b) (c) (d) (e)

Figure 2. Example of barcode denoising for the isotropic and anisotropic models on a 175×175 image. a) Clean image. b) Noisy image. c) Anisotropic denoising. d) Isotropic denoising. e) Median filter.

of the original image. The penalty parameter λ for each model was chosen so as to maximize the on the 300×300 image. As expected, the anisotropic TV regularizer is more appropriate for the underlying structure of the image, and thus obtains lower reconstruction errors. An example is shown in Figure 2, where we also observe visually better reconstructions via the anisotropic model. Additional experimental results are in (Barbero & Sra, 2011).

Regarding running times, our PN solver vastly outperforms Coordinate Optimization and anisotropic PDHG. The isotropic version of the problem is simpler than the anisotropic one, so it is no surprise that the carefully tuned PDHG approach requires less time than PN. It is also worth mentioning that in (Choksi et al., 2010) an ℓ_1 loss is used, and denoising cast as a Linear Program, to which a generic solver is applied; this approach requires runtimes of over 10^3 seconds for the largest image.

5.3. Image deconvolution

With little added effort our two-dimensional TV solver can be employed for the harder problem of image deconvolution, which takes the form

$$\min_{\boldsymbol{x}} \quad \tfrac{1}{2}\|\boldsymbol{K}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda R(\boldsymbol{x}),$$

where K is a blur operator, R is a regularizer, and $\boldsymbol{x}$ encodes an image. As stated before, this problem can also be solved using prox_R as a building block. Precisely this is done by the solver SALSA (Afonso et al., 2010). We plug our 2D-TV solver directly into SALSA to obtain a fast anisotropic deconvolution algorithm. Table 5 presents numerical results (visual results are in (Barbero & Sra, 2011)) for deconvolution of noisy barcode images subjected to motion blurring. Comparing against SALSA's default isotropic denoising operator, again an anisotropic model produces a better reconstruction. Results for Richardson-Lucy (RL) (Biggs & Andrews, 1997) as implemented in Matlab are also presented, showing much faster filtering times but inferior reconstruction quality.

Table 5. Deconvolution results for anisotropic and isotropic models using the SALSA solver, and MATLAB's Richardson-Lucy (RL) method. ISNR (dB) values and runtimes (in secs) are shown.

	ANISOTROPIC		ISOTROPIC		RL	
	ISNR	TIME	ISNR	TIME	ISNR	TIME
100	1.55	1.19	1.10	0.12	0.73	0.04
175	2.79	0.81	2.15	0.55	0.79	0.18
300	4.07	3.34	3.07	2.40	1.07	0.46
375	4.05	5.41	2.92	3.71	1.13	0.61
500	3.21	8.98	2.37	5.71	1.04	1.26

Acknowledgments

With partial support from Spain's TIN 2010-21575-C02-01 and FPU–MEC grant reference AP2006–02285.