COM-514 Project - Diffuser Cam

Adrian Jarret (331279), Eelis Mielonen (323681), Mehmet Onurcan Kaya (336453), Kaan Okumus (335931)

I. Introduction

In this project we try to perform lensless imaging by using a piece of tape as the encoding element of a digital camera. Such a technique allows to circumvent the use of a regular converging lens, obtaining at the same time more compact and lighter devices. The drawback of removing the lens is that the measured images are highly corrupted and most of the times not even understandable by human eyes. To reconstruct clean images, we need appropriate numerical treatment of the measured data. This is addressed by formulating inverse problems to solve for the ground truth images. In this project we study the reconstruction results obtained by various cost functionals and optimisation algortihms, leveraging the flexibility of the pycsou python library.

Our first step is to measure the *point spread function (PSF)* of the DiffuserCam built with a piece of tape. We are then able to model the linear measurement operator $\boldsymbol{H}$ associated to the camera with this PSF, leading to inverse problems $\boldsymbol{y} \approx \boldsymbol{H}\boldsymbol{x}$ where $\boldsymbol{y}$ is the measured data and $\boldsymbol{x}$ is the image to reconstruct. In this document, we present some reconstruction methods relying on different priors and consequently solving algorithms. More specifically, we propose a performance analysis and benchmarks of these methods on two mini datasets. ADMM is compared with a simple LASSO solved with APGD, and a version of that latter problem with a sparsity constraint on the gradient in the image, ie. we replace the ℓ_1 norm with the *Huber norm* of the finite difference operator applied to x.

II. Analysis of PSF

We measure the empirical PSF of the DiffuserCam, the result is provided in Figure 1 for grayscale and Figure 2 for RGB version. The exposure value and red/blue gains are adjusted such that pixel values in the PSF image are not saturated and there is no clipping (the full dynamic range is used). Also, the Bayer gains are adjusted for white-balancing since we used a white point source. Without such an adjustment the result would be greenish since the Bayer pattern dedicates more pixels to green than to red and blue.

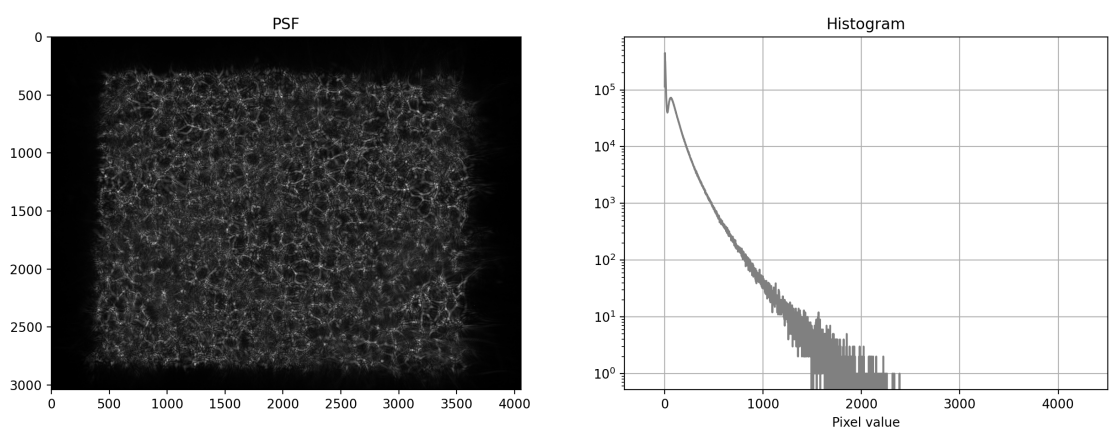


Fig. 1: Grayscale PSF and histogram of pixels values.

For the DiffuserCam, we want a PSF with large support and many directional filters. When our diffuser is "focused", we will have thin lines (high contrast contours), which means that small shifts will be uncorrelated with each other. This means convolution with such a PSF will act as directional filter. The width of the PSF's autocorrelation peak (see Figure 3) helps to determine if the PSF has these properties.

When illuminated by a point source, the convex bumps on the diffuser concentrate rays, creating a high contrast caustic pattern (PSF) at the sensor. If the PSFs of neighboring point sources are very similar, it is challenging to distinguish between the sources. This causes poor reconstruction quality, lowering the SNR and resolution. To quantify this, we can consider two point sources located at different locations and measure their similarity by the 2D autocorrelation of the PSF. We quantify the autocorrelation sharpness by looking at its -3dB width. Thus, the resolution of the reconstruction will be better if this width is smaller in the PSF. From 3, we expect to have better reconstructions in the red channel with respect to other channels.

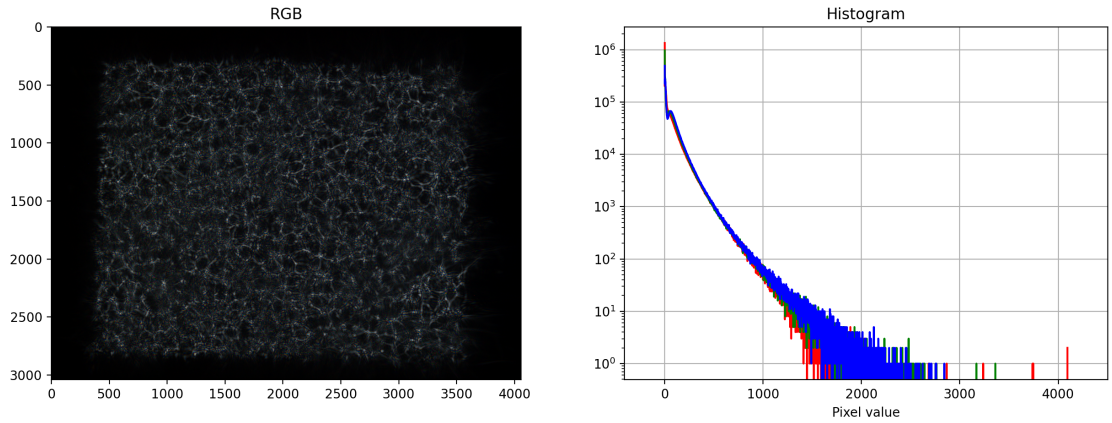


Fig. 2: Color PSF and histogram of RGB pixels values.

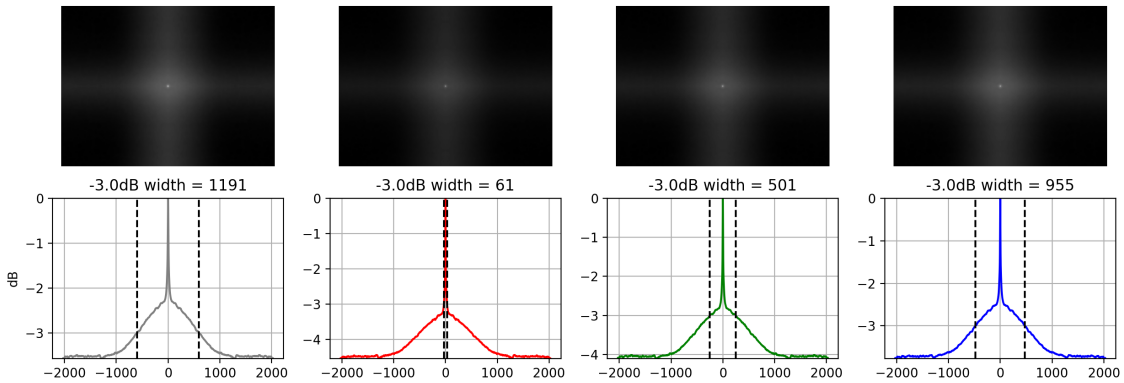


Fig. 3: Autocorrelation analysis of the PSF.

III. Methods and their Reconstruction Results

The experimental setup consists in acquiring data from an image displayed on a digital monitor 40 centimeters away from the DiffuserCam. Such data are identified with a measurements vector $\boldsymbol{y} \in \mathbb{R}^N$. The measurement procedure is modelled as a 2-dimensional convolution with the previously measured PSF under the linear operator $\boldsymbol{H} \in \mathbb{R}^{N \times N}$. Our goal is to reconstruct a source image $\hat{\boldsymbol{x}} \in \mathbb{R}^N$ explaining the data. To do so, we formulate various optimization problem with least-squares data fidelity and different penalties, accounting for different priors in the reconstructed images.

In order to establish a uniform comparison of the algorithms and priors, we decided to compare the iterations and reconstructions obtained at a certain accuracy threshold of 10^{-4} on the relative improvement of the iterates. We also compare them with their converged outputs.

A. Penalized Least-Square Problem with Tikhonov Regularization (Ridge Regression)

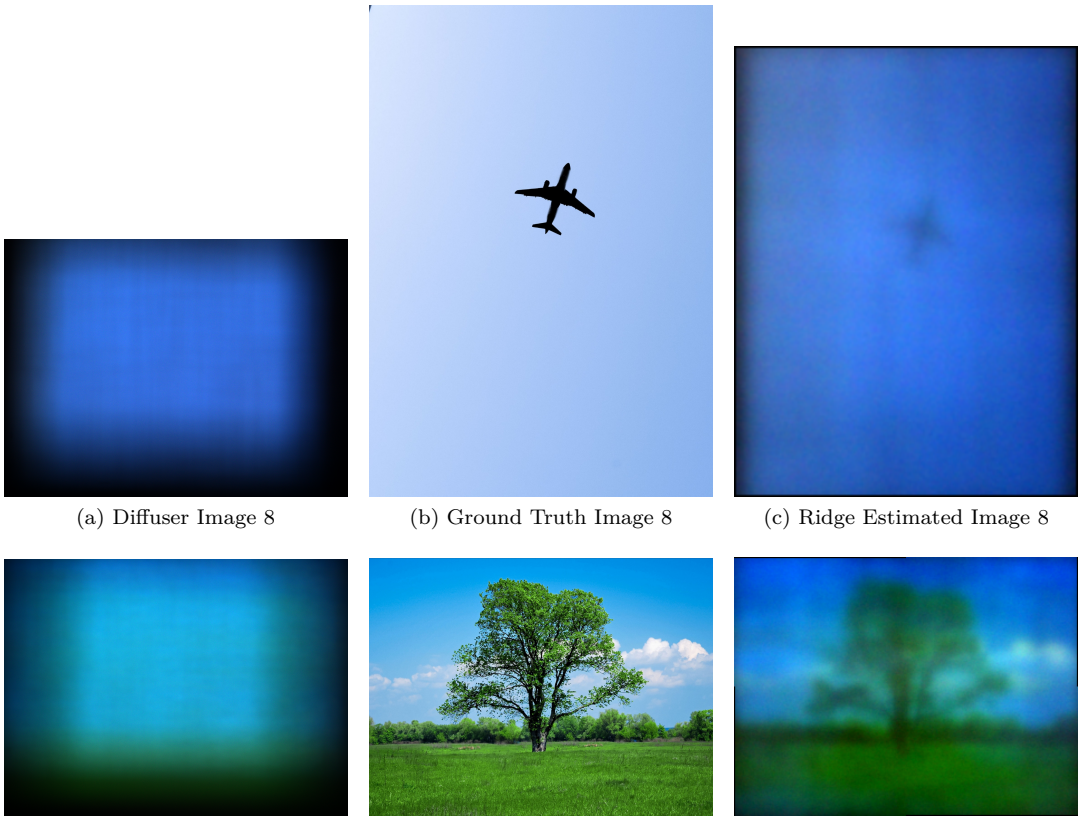
The Tikhonov regularization, also known as ridge regression, is formalized as follows:

$$\min_{\boldsymbol{x} \in \mathbb{R}^N} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2^2 + \lambda \|\boldsymbol{x}\|_2^2 \tag{1}$$

By introducing a differentiable penalty, this approach has the advantage of preserving a differentiable cost functional, allowing the use of regular gradient methods. It is known to promote smooth reconstructed images that might seem "blurry" in most of the cases. We observed that behaviour in our numerical experiments, that we provide in Figure 4.

B. Penalized Least-Square Problem with ℓ_1 Regularization (LASSO Problem)

Another common approach consists in using the ℓ_1-norm instead of the ridge regression ℓ_2-norm in the penalty term. The resulting optimization problem (2), coined *LASSO*, is known for promoting sparsity in the solutions. The



(a) Diffuser Image 8 (b) Ground Truth Image 8 (c) Ridge Estimated Image 8

(d) Diffuser Image 2 (e) Ground Truth Image 2 (f) Ridge Estimated Image 2

Fig. 4: Reconstruction results of ridge estimation for two different images.

non-differentiability of the regularizer requires to use adapted solving algorithms, in our experiments we rely on the proximal algorithm APGD.

$$\min_{\boldsymbol{x}\in\mathbb{R}^N} \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{x}\|_1 \tag{2}$$

Our reconstructions are illustrated in Figure 5. We observed that the sparsity promoting prior leads to sharper images in few cases, with more contrast than with the ridge reconstruction, however on average most reconstructions suffered from blurriness.

C. Nonnegative Least-Square Problem

To ensure that the proposed solution of the optimization problem is an actual image, it is interesting to enforce a positivity constraint, leading to the non-negative least-squares problem (3).

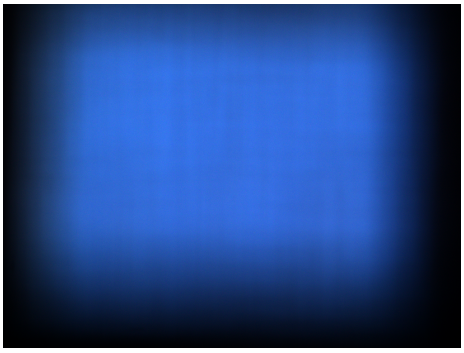
$$\min_{\boldsymbol{x}\in\mathbb{R}_+^N} \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{x}\|_2^2 \tag{3}$$

This problem can be reformulated as in problem (4) using an additive indicator function

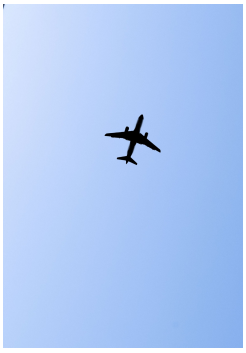
$$\min_{\boldsymbol{x}\in\mathbb{R}^N} \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{x}\|_2^2 + \iota_{[0,\infty]}(\boldsymbol{x}) \tag{4}$$

where $\iota_{[0,\infty]}(\boldsymbol{x}) = \begin{cases} 0 & \text{if} \quad \boldsymbol{x}\in[0,\infty]^N \\ +\infty & \text{otherwise} \end{cases}$

As $\iota_{[0,\infty]}$ is proximable, we can still solve the problem (4) using APGD. Running this algorithm to the same accuracy threshold as before does not yield clear differences in terms of metrics. Visually the reconstructions look very similar than regular ridge regression, slightly less blurry for some examples. We provide some of our results in Figure 6.



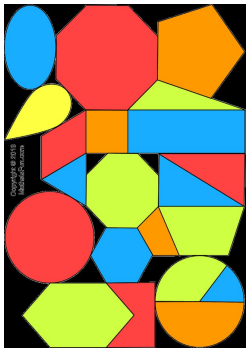
(a) Diffuser Image 1



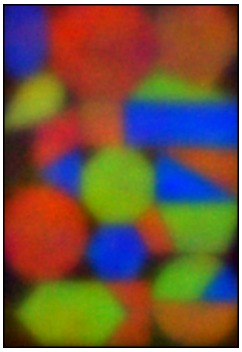
(b) Ground Truth Image 1

(c) LASSO Estimated Image 1

(d) Diffuser Image 2



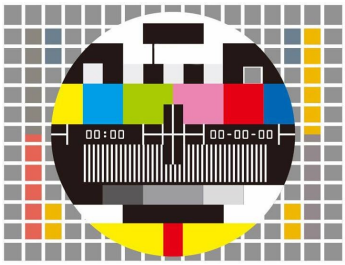
(e) Ground Truth Image 2



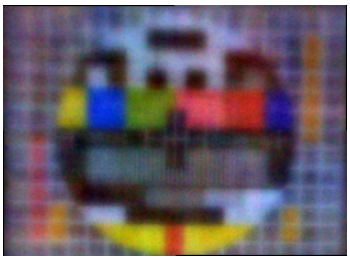
(f) LASSO Estimated Image 2

Fig. 5: Reconstruction results of LASSO estimation for two different images.

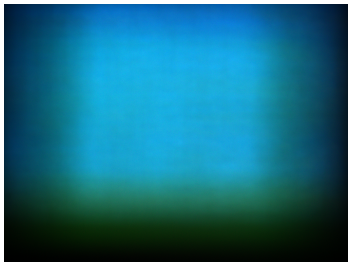
(a) Diffuser Image 1



(b) Ground Truth Image 1



(c) NNLS Estimated Image 1



(d) Diffuser Image 2

(e) Ground Truth Image 2

(f) NNLS Estimated Image 2

Fig. 6: Reconstruction results of NNLS estimation for two different images.

Representer Theorem for nonnegativity constraints states that our optimization problem admits a unique L-sparse solution for every $\boldsymbol{y} \in \mathbb{R}^L$. But since our measurement operator $\boldsymbol{H} \in \mathbb{R}^{L \times L}$, this sparsity does not imply anything for the reconstructed image.

D. Generalized LASSO Problem with Discrete Cosine Transform

By introducing a linear operator inside the ℓ_1 penalty term of the LASSO problem, it is possible to look for solutions that are not necessarily sparse by themselves but sparse in a certain transform domain. This operation leads to the so-called *generalized LASSO* problems. In our experiments, we focus on the case of reconstructing images with a sparse *discrete cosine transform (DCT)*, *i.e.* with only a few non-null DCT coefficients. To do so, we use the following optimization problem

$$\min_{\boldsymbol{x} \in \mathbb{R}^N} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2^2 + \lambda \|\boldsymbol{x}\|_1 \tag{5}$$

where the image x is now assumed to be $x = \mathrm{DCT}_2(\boldsymbol{X})$, where DCT_2 is the 2D Type-2 DCT operator and $\boldsymbol{X}$ are the DCT coefficients of x. As DCT_2 is an orthonormal operator of $\mathbb{R}^{N \times N}$, the inverse operator is given by the adjoint operator

$$\mathrm{DCT}_2^{-1} = \mathrm{DCT}_2^*,$$

such that this operation can easily be implemented as a subclass of `LinearOperator` (from `pycsou.core.linop`).

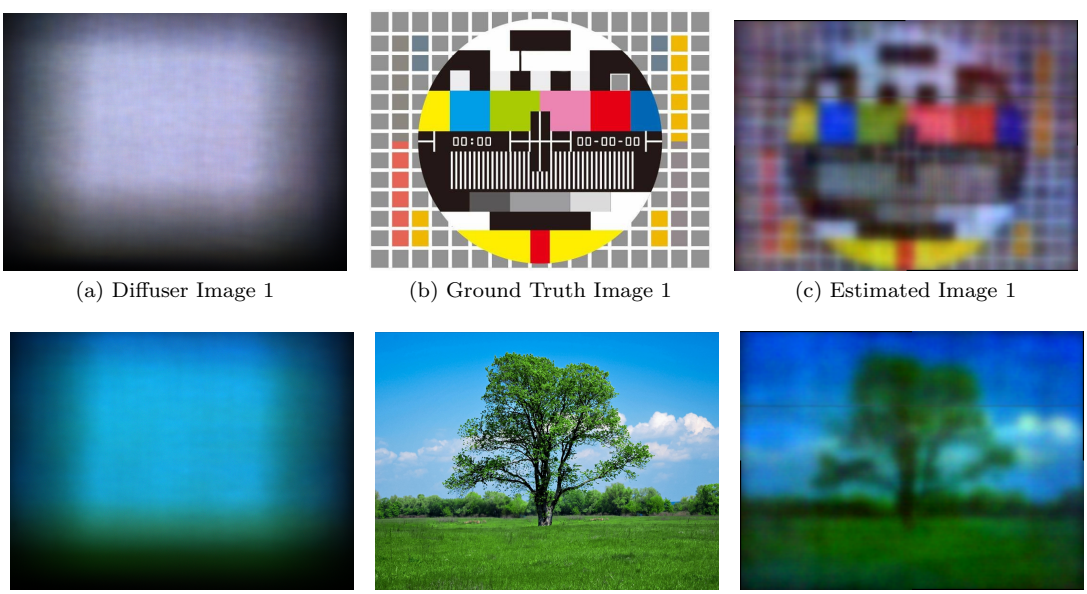
With the following change of variable

$$\boldsymbol{X} = \mathrm{DCT}_2(\boldsymbol{x}),\ \boldsymbol{x} = \mathrm{DCT}_2^{-1}(\boldsymbol{X}),$$

it is possible to rewrite the generalized problem (5) as the following regular LASSO problem

$$\min_{\boldsymbol{X} \in \mathbb{R}^N} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{H}_{\mathrm{DCT}}(\boldsymbol{X})\|_2^2 + \lambda \|\boldsymbol{X}\|_1 \tag{6}$$

with $\boldsymbol{H}_{\mathrm{DCT}} = \boldsymbol{H} \circ \mathrm{DCT}_2^{-1}$ still being a linear operator. For the most of the natural images, high frequency components in the DCT domain is very weak since there are a few sudden changes. Thus, this approach might be of particular interest for natural images that tend to have sparse or nearly sparse DCT transform. We provide some of our reconstructions in Figure 7.



(a) Diffuser Image 1 (b) Ground Truth Image 1 (c) Estimated Image 1

(d) Diffuser Image 2 (e) Ground Truth Image 2 (f) Estimated Image 2

Fig. 7: Reconstruction results of Gen. LASSO with DCT penalty for two different images.

From the Representer Theorem of ℓ_1 Regularization, we know that the optimization problem $\min_{\boldsymbol{x} \in \mathbb{R}^N} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2^2 + \lambda \|DCT(\boldsymbol{x})\|_1$ has a unique solution $\boldsymbol{\alpha}^\star = DCT^{-1}\boldsymbol{\beta}_K^\star$ for some K-sparse vector $\boldsymbol{\beta}_K^\star$, $K \leq N$. Thus, we have a sparse reconstruction in the DCT domain, which assumably would result in smooth natural images. Possibly due to measurement noise, this prior didn't provide a noticeable improvement to the previous approaches.

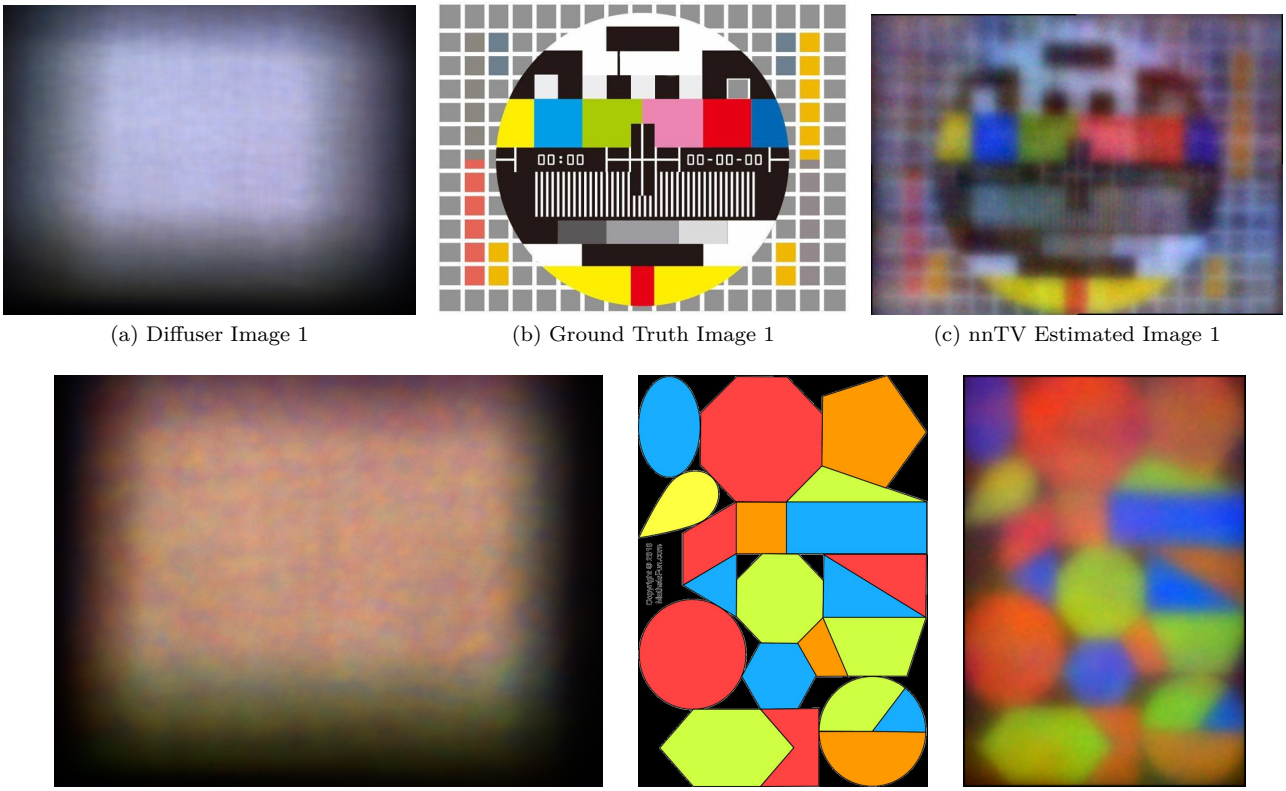
E. Least-Squares Regression with Total Variation Penalty and Non-Negative Prior

Another commonly used generalized problem relies on the *Total Variation (TV)* penalty, as provided in problem (7) in addition to a positivity constraint, introduced as before with the indicator function of the non-negative orthant.

$$\min_{\boldsymbol{x}\in\mathbb{R}^N} \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{x}\|_2^2 + \iota_{[0,\infty]}(\boldsymbol{x}) + \lambda\|\boldsymbol{D}\boldsymbol{x}\|_1, \tag{7}$$

$\boldsymbol{D}$ is finite-difference 2D gradient operator. By promoting sparse discrete gradient, *i.e.* sparse "jumps", the TV regularizer works well for reconstructing piecewise constant images.

As formalized in problem (7), it is no longer possible to directly apply APGD, we thus rely on a *Primal Dual Splitting method (PDS)*. Although it requires significantly more iterations than APGD, PDS is able to handle such a sum of two proximable penalty terms. Some reconstructions are provided in Figure 8. In particular, the subfigure (f) is very interesting, as it enlights the reconstruction capacity on a piecewise constant image.



(a) Diffuser Image 1 (b) Ground Truth Image 1 (c) nnTV Estimated Image 1

(d) Diffuser Image 2 (e) Ground Truth Image 2 (f) Estimated Image 2

Fig. 8: Reconstruction results with nnTV penalty for two different images.

F. Penalized Least-Square Problem with Huber Norm and Nonnegative Prior Regularization

In order to speed up the solving time for the previous nonnegative TV penalized problem (7), a strategy consists in using a *smoothed* ℓ_1-norm known as the *Huber norm*. This leads to the resulting problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^N} \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{H}\boldsymbol{x}\|_2^2 + \iota_{[0,\infty]}(\boldsymbol{x}) + \lambda\|\boldsymbol{D}\boldsymbol{x}\|_{H,\delta}, \tag{8}$$

where the Huber norm is defined as

$$\|\boldsymbol{z}\|_{H,\delta} = \sum_{m=1}^{M} h_\delta(z_m), \qquad \boldsymbol{z}\in\mathbb{R}^M$$

using the Huber function $h_\delta : \mathbb{R} \to \mathbb{R}_+$ of parameter $\delta > 0$ as

$$h_\delta(z) = \begin{cases} \frac{1}{2}z^2 & \text{for } |z| < \delta \\ \delta(|z| - \delta/2) & \text{otherwise.} \end{cases}$$

The Huber function is differentiable with continuous derivative, even at breakpoints $|z| = \delta$, its derivative is given by

$$h'_\delta(z) = \begin{cases} z & \text{for } |z| < \delta \\ \delta \operatorname{sgn}(z) & \text{otherwise.} \end{cases}$$

Consequently, the Huber norm is diferentiable with gradient given by

$$\nabla \|\boldsymbol{z}\|_{H,\delta} = \left(h'_\delta(\boldsymbol{z}_m)\right)_{m=1,\ldots,M} \in \mathbb{R}^M,$$

which leads to

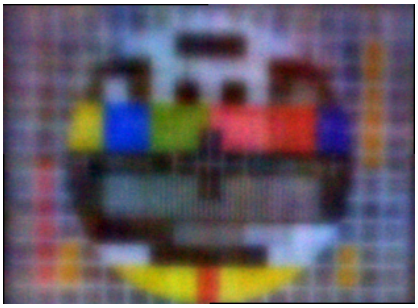
$$\left\|\nabla \|\boldsymbol{z}_1\|_{H,\delta} - \nabla \|\boldsymbol{z}_2\|_{H,\delta}\right\|_2 \leq \|\boldsymbol{z}_1 - \boldsymbol{z}_2\|_2, \quad \text{for } \boldsymbol{z}_1, \boldsymbol{z}_2 \in \mathbb{R}^M. \tag{9}$$

Equation (9) ensures that the Huber norm has 1-Lipschitz continuous gradient, useful to tune the β parameter of APGD.

Some reconstruction are provided in Figure 9. As this method relies on the same piecewise constant prior as the TV penalty, it also performs well for that type of images, as we can see in subfigure (f). Perceptually, it seems to be even better than the image obtained in Figure 8.

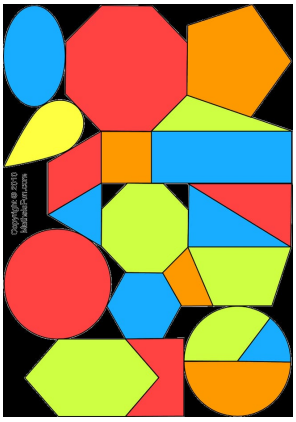
(a) Diffuser Image 1

(b) Ground Truth Image 1

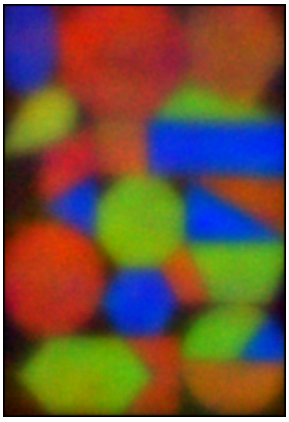


(c) Estimated Image 1

(d) Diffuser Image 2



(e) Ground Truth Image 2

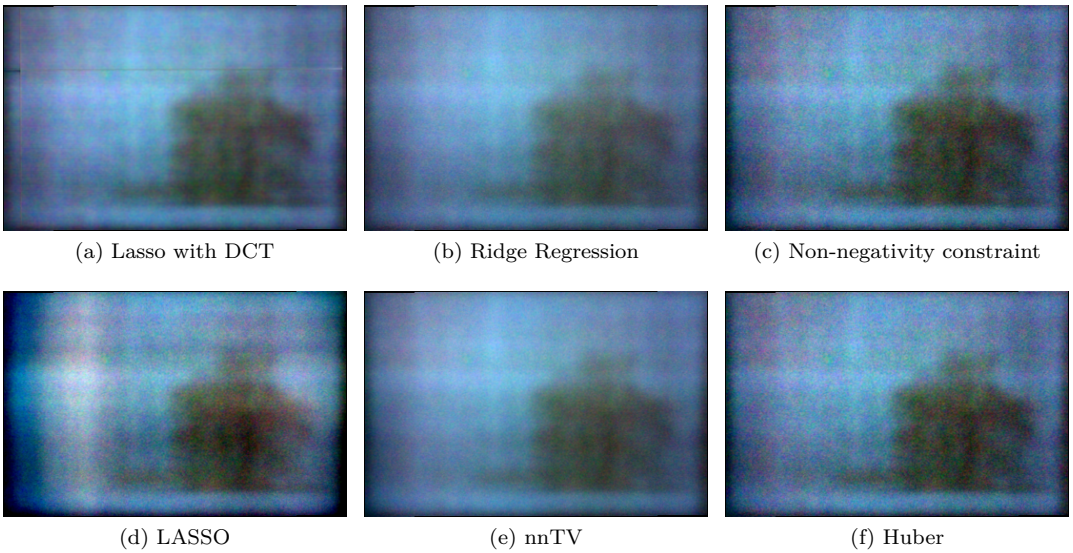


(f) Estimated Image 2

Fig. 9: Reconstructions using Huber norm approximation of TV penalty for two different images.

G. Reconstruction of Low Contrast Images

The reconstructions over many of the images were similar across the algorithms, but one special case stood out, that we emphasize in this section. When dealing with low contrast images, like image 5 and 7 from out mini dataset, LASSO and ADMM performed noticeably worse (see Figure 10, and appendix), a behaviour that differs from what we observed with the other images. In this special case, reconstruction with sparse DCT coefficients seems to produce the best image.



(a) Lasso with DCT (b) Ridge Regression (c) Non-negativity constraint

(d) LASSO (e) nnTV (f) Huber

Fig. 10: Reconstructions of a low contrast image for all methods.

IV. Benchmark of Different Approaches

We chose to measure the CPU time it takes for LASSO, Huber, and ADMM to reach a reconstruction of a similar quality on a subset of the Flicker data-set. The subset consisted of 3 images, namely images 10-12. To achieve such a benchmark, we stopped our algorithms at the same accuracy threshold of $5e^{-4}$ of relative improvement, and hand-tuned the iterations of ADMM until we got a reconstruction of a similar quality, in terms of average PSNR, SSIM, and MSE. We provide the benchmark results in Table II.

We see that ADMM reaches this reconstruction goal far earlier than the APGD based approaches, and that using a more sophisticated Huber functional mainly adds preparation time without speeding up convergence. This table not describe the final reconstruction quality that can be achieved by the algorithms, but rather how quickly a reasonable approximation can be made.

We also ran a benchmark of the different methods over our mini-dataset. We observed that PDS-based algorithms are much slower than the APGD-based ones, even though that latter have required more than 500 iterations to obtain satisfying reconstructions. The different priors used do not lead to significant variations in terms of the studied metrics for this dataset.

TABLE I: Average reconstruction benchmarks for different approaches for our mini-dataset consisting of 11 images. All algorithms were executed until they converge.

Reconstruction Method	Number of Iterations	Set-up Time (s)	Processing Time (s)	MSE	PSNR	SSIM
ADMM	10	13.62	10.68	0.14	8.78	0.35
Ridge Regression	650	14.97	415.03	0.07	12.28	0.51
LASSO	1000	14.52	604.61	0.09	10.82	0.45
NNLS	1000	14.50	597.79	0.08	11.31	0.45
Huber	1000	24.59	786.43	0.08	11.31	0.45
DCT	1000	16.14	802.43	0.07	11.93	0.51
nnTV	2000	23.28	19750.18	0.07	12.11	0.50

TABLE II: Average reconstruction performances of different algorithms for MIR Flickr dataset.

Reconstruction Method	MSE	SSIM	PSNR	LPIPS	CPU time (s)
ADMM	0.1803	0.3575	7.8099	0.7796	9.65
LASSO	0.1861	0.4016	7.7935	0.6711	122.82
Huber	0.1935	0.3944	7.6016	0.6733	147.51

TABLE III: Average reconstruction benchmarks for different approaches for our mini-dataset consisting of 11 images. All algorithms were stopped at the same accuracy threshold on relative improvement and used the same regularization parameter.

Reconstruction Method	Set-up Time (s)	Processing Time (s)	MSE	PSNR	SSIM
ADMM	13.62	10.68	0.14	8.78	0.35
Ridge Regression	14.97	415.03	0.07	**12.28**	0.51
LASSO	14.52	**266.73**	0.07	11.85	0.52
NNLS	**14.50**	263.09	0.07	11.99	0.52
Huber	24.59	346.91	0.07	11.99	0.52
DCT	16.14	372.18	0.07	12.22	0.52
nnTV	23.28	310.27	0.09	11.15	0.52

V. Discussion

As a first remark, without analysing in detail each reconstruction, we should notice that the numerical treatment proposed to correct the physical effects of the tape as converging device of the DiffuserCam leads to actual images, that can easily be recognized by human eye. We provide in appendix a summary view of all of our images and obtained reconstructions.

From these experiments it is hard to draw conclusions about the visual effect that different priors have on reconstructions, however with our experimental set up we were still able to compare the time required by each approach to reach a certain accuracy threshold. We found that ADMM is by far the fastest method (when observing the reconstruction performances on the Flickr dataset), and out of the approaches that we implemented, Ridge regression performed as well as the other more complicated approaches. When solving the LASSO problem we see that imposing sparsity in the DCT domain gives a slightly better reconstruction in terms of PSNR (Table III). Solving the LASSO problem in the DCT domain also takes slightly less time to converge to the same accuracy as Ridge Regression (with the same lambda and accuracy threshold).

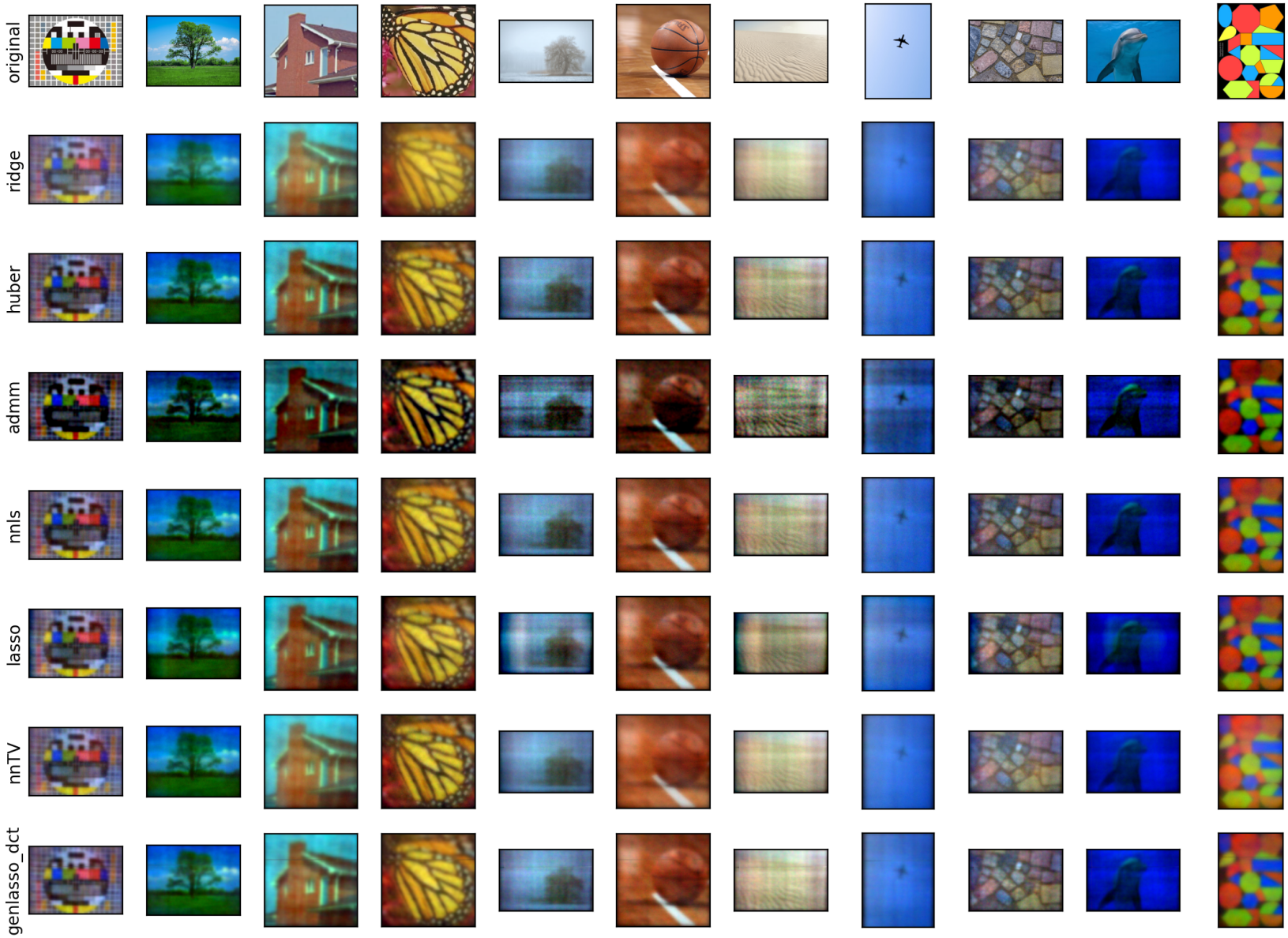
One interesting thing to note is that even though ADMM was faster, it performed worse than all the approaches we tried on a low contrast images (Image 5 and 7, check appendix).

Commenting on the visual results is difficult, but it is clear that using a sparsity promoting prior with LASSO improved the results over ridge regression for image 1, which is a sparse image (one dark plane with a blue sky background). In order to improve the study, and to properly study the effect of regularizers, a sweep over the value of λ could be performed to find the best possible reconstructions. Note that the value of λ depends on each image and should be tuned independently.

Another limitation with regards to computing metrics was the lack of access to lensed ground truth images. In a future study both lensed and diffused images could be captured to have a more accurate ground truth comparison. On the other hand measurements could also be simulated by convolving the test images directly with a measured psf.

Appendix

A. Results of Reconstruction Methods for the Whole Mini Data-set



To create this Figure, we specified the convergence points for each reconstruction method and got the estimated images according to these converged points.