Putting MAP back on the map

Patrick Pletscher[1], Sebastian Nowozin[2], Pushmeet Kohli[2], and Carsten Rother[2]

[1] ETH Zurich, Switzerland
[2] Microsoft Research Cambridge, UK

The supplementary material mainly contains additional results which did not fit into the paper, due to space limitations.

1 Implementation details

Inference For MAP inference we use the TRW-S implementation bundled in the Middlebury MRF suite. For MMSE inference we implemented a Gibbs sampler with a burn-in of 100 iterations and another 100 iterations used to compute the marginals for the MMSE prediction. The MAP inference was observed to be substantially faster than MMSE.

Learning The MPLE and the max-margin training are our own implementations. The MPLE is computed using the Matlab `minFunc` L-BFGS solver. The evaluation of the MPL objective and its gradient are implemented in C++ for speed reasons. The max-margin QP objective is constructed from within Matlab and solved using the MOSEK solver. The maximally violating labelings are computed using TRW-S inference. We cycle through the data and solve the QP anew after each inference step. As the number of constraints grows with each iteration of the max-margin training algorithm, we prune the least binding constraints of the QP. This is done fairly conservatively in order to not sacrifice the accuracy of the solver. For all our experiments we chose $\lambda = 10^{-3}$ for max-margin and no regularizer for the MPLE. We do not see dramatic changes of the max-margin results when changing λ. However, choosing it too small results in numerical difficulties when solving the max-margin objective. For MPLE we did not observe numerical problems when choosing no regularizer. Also, we did not see improvements of the results when including a regularizer. As we only have few parameters to estimate and a lot of data we do not expect the regularizer to play a crucial role.

2 Consistency experiment for the synthetic data

In order to show the consistency of the MPLE, the difference between the true and the estimated weights is investigated. If the estimator is consistent, then for large N, their difference should become negligible. There is one technical problem with this comparison as the (non-regularized) MPLE is not unique: there exist infinitely many weights with the same pseudo-likelihood, for more details see below. In order to make the weights comparable, we subtract for both, unary and pairwise the minimum weight to obtain a normalized weight:

$$\bar{\boldsymbol{w}} = [(\boldsymbol{w}^u)^\mathsf{T} - \min_k w_k^u, (\boldsymbol{w}^p)^\mathsf{T} - \min_k w_k^p]^\mathsf{T}.$$

The difference of two parameters is then assumed to be given by the normalized Euclidean distance of the two normalized weight vectors:

$$d(\boldsymbol{w}^{true}, \boldsymbol{w}) = \|\bar{\boldsymbol{w}}^{true} - \bar{\boldsymbol{w}}\|_2 / \|\bar{\boldsymbol{w}}^{true}\|_2.$$

Fig. 1 shows that the MPLE indeed converges to the true parameters. Furthermore we also investigate on the MSE of the prediction for different estimation and prediction strategies. We observed that MPLE/MMSE outperforms MM/MAP, while MPLE/MAP is significantly worse (start point 5.2 MSE, end point 13.3 MSE; not shown for readability). It is not surprising that MPLE/MMSE leads to better accuracy here, as after all it is the optimal strategy according to Bayesian decision theory (assuming MPLE converged to MLE).

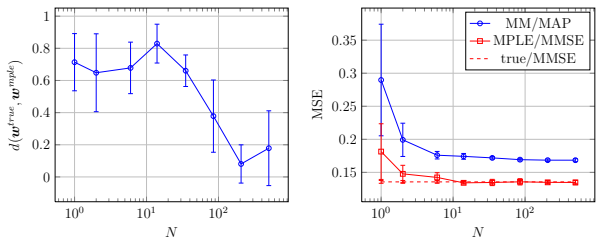


Fig. 1: Consistency of pseudo-likelihood for synthetic data. Left: The difference between the estimated weights and the true weights becomes small for large N. Right: Prediction error of the different methods for varying data set sizes. In addition we also show the prediction error with the data generating parameters.

2.1 Non-uniqueness of M(P)LE

We prove here the non-uniqueness for unregularized maximum likelihood estimation. All the computations also carry over to maximum pseudo-likelihood estimation. The log-likelihood is given by:

$$\log P(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w}) = \langle \boldsymbol{w}, \boldsymbol{s}(\boldsymbol{x},\boldsymbol{y})\rangle - \log \sum_{\boldsymbol{y}'} \exp(\langle \boldsymbol{w}, \boldsymbol{s}(\boldsymbol{x},\boldsymbol{y}')\rangle).$$

Consider the log-likelihood of the same weight vector where the unary and pairwise weights are all shifted by a constant c^u and c^p, respectively. Let us denote the resulting weight vector by $\widetilde{\boldsymbol{w}}$. The log-likelihood is given by

$$\begin{aligned}
\log P(\boldsymbol{y}|\boldsymbol{x},\widetilde{\boldsymbol{w}}) &= \langle \boldsymbol{w}^u + c^u, \boldsymbol{s}^u(\boldsymbol{x},\boldsymbol{y})\rangle + \langle \boldsymbol{w}^p + c^p, \boldsymbol{s}^p(\boldsymbol{y})\rangle \\
&\quad - \log\Bigg(\sum_{\boldsymbol{y}'} \exp\big(\langle \boldsymbol{w}^u + c^u, \boldsymbol{s}^u(\boldsymbol{x},\boldsymbol{y}')\rangle + \\
&\qquad \langle \boldsymbol{w}^p + c^p, \boldsymbol{s}^p(\boldsymbol{y}')\rangle\big)\Bigg) \\
&= \langle \boldsymbol{w}, \boldsymbol{s}(\boldsymbol{x},\boldsymbol{y})\rangle + c^u + c^p \\
&\quad - \log\Bigg(\sum_{\boldsymbol{y}'} \exp\big(\langle \boldsymbol{w}, \boldsymbol{s}(\boldsymbol{x},\boldsymbol{y})\rangle\big)\Bigg) \\
&\quad - c^u - c^p \\
&= \log P(\boldsymbol{y}|\boldsymbol{w},\boldsymbol{w}).
\end{aligned}$$

The second step follows from the fact that the elements of the sufficient statistics $\boldsymbol{s}^u(\boldsymbol{x},\boldsymbol{y}')$ and $\boldsymbol{s}^p(\boldsymbol{y}')$ are non-negative and sum up to the same constant for all the label $\boldsymbol{y}'$.

3 Synthetic Experiments

Fig. 2 shows the extended results of the same experiment as in the paper. Fig. 3 shows an experiment with the same form of the ground-truth potentials as the first experiment. However, the pairwise potentials are chosen weaker. Here, MPLE/MAP performs closer to the other methods than in the first experiment. Fig. 4 shows a choice of the ground-truth potentials where the unary term has a different shape, in contrast to the previous experiment. In this case MPLE does not identify the true parameters, thus both, MPLE/MAP and MPLE/MMSE are worse than MM/MAP. This illustrates the deficiencies of pseudo-likelihood even for relatively large data sets. We expect that with more data the MPLE would identify the true parameters. Finally, Fig. 5 shows a configuration where the potential weights are different to the previous experiment. In this case, the MSE goes down for MM/MAP when increasing the misspecification, i.e. ϵ. Looking at the ground-truth samples ($\boldsymbol{x}$ and $\boldsymbol{y}$) (left images in 3rd and 4th row), we see that the labels become smoother for larger ϵ. Such a behavior can be achieved with the standard pairwise model without the longer range connections. This shows the power of max-margin training.

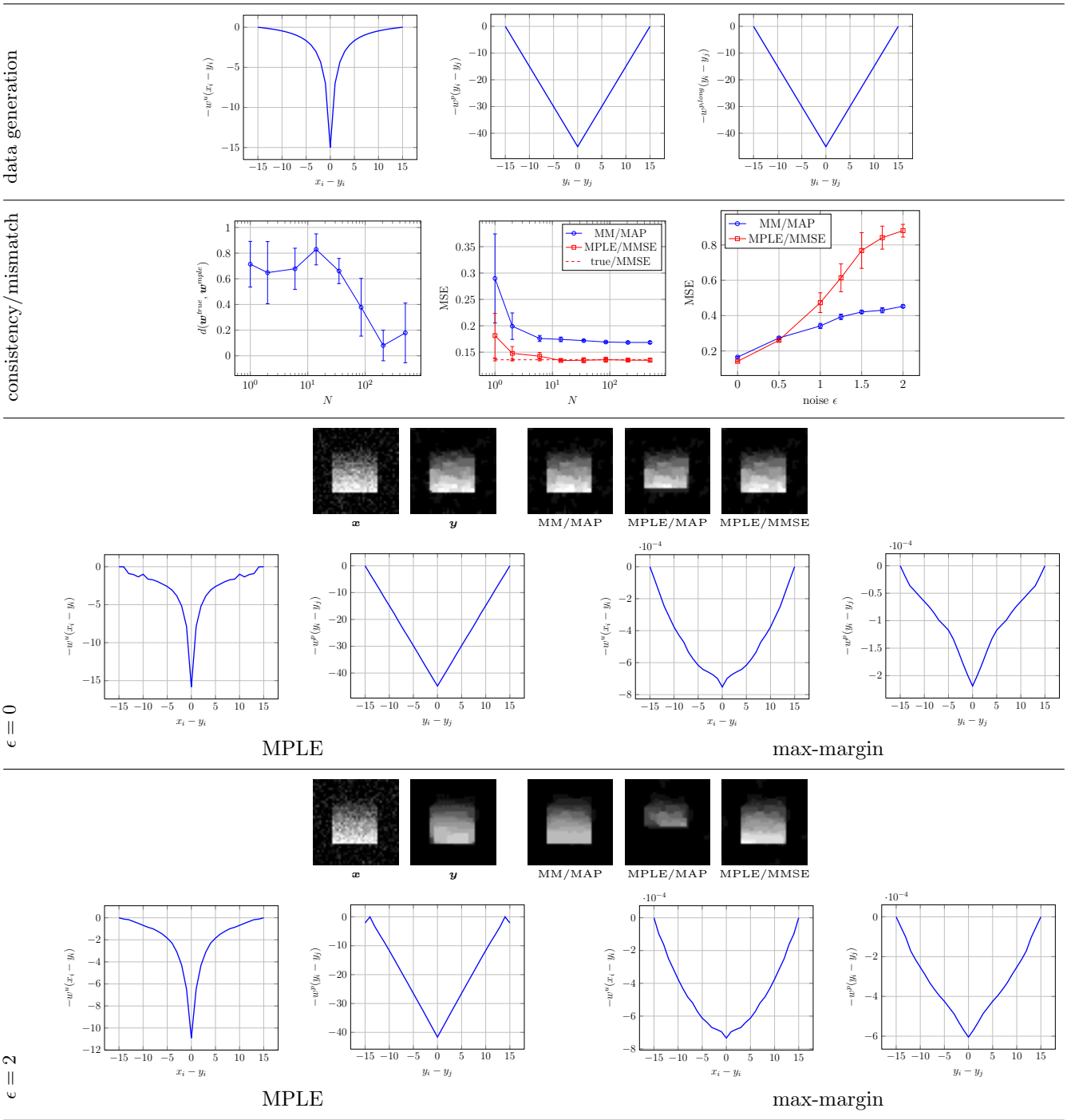


Fig. 2: First row: Weights used to generate the data: unary(left), pairwise - short distance (middle), pairwise - long distance (right). Second row: consistency of the MPLE (left); MSE result without misspecification, i.e. $\epsilon = 0$, (middle); and results for increased level of misspecification (right). Note, MPLE/MAP is not shown in the plots, as the MSE is much larger. Third row: Results for the case of no misspecification ($\epsilon = 0$). Last row: Results for the case of misspecification ($\epsilon = 2$).

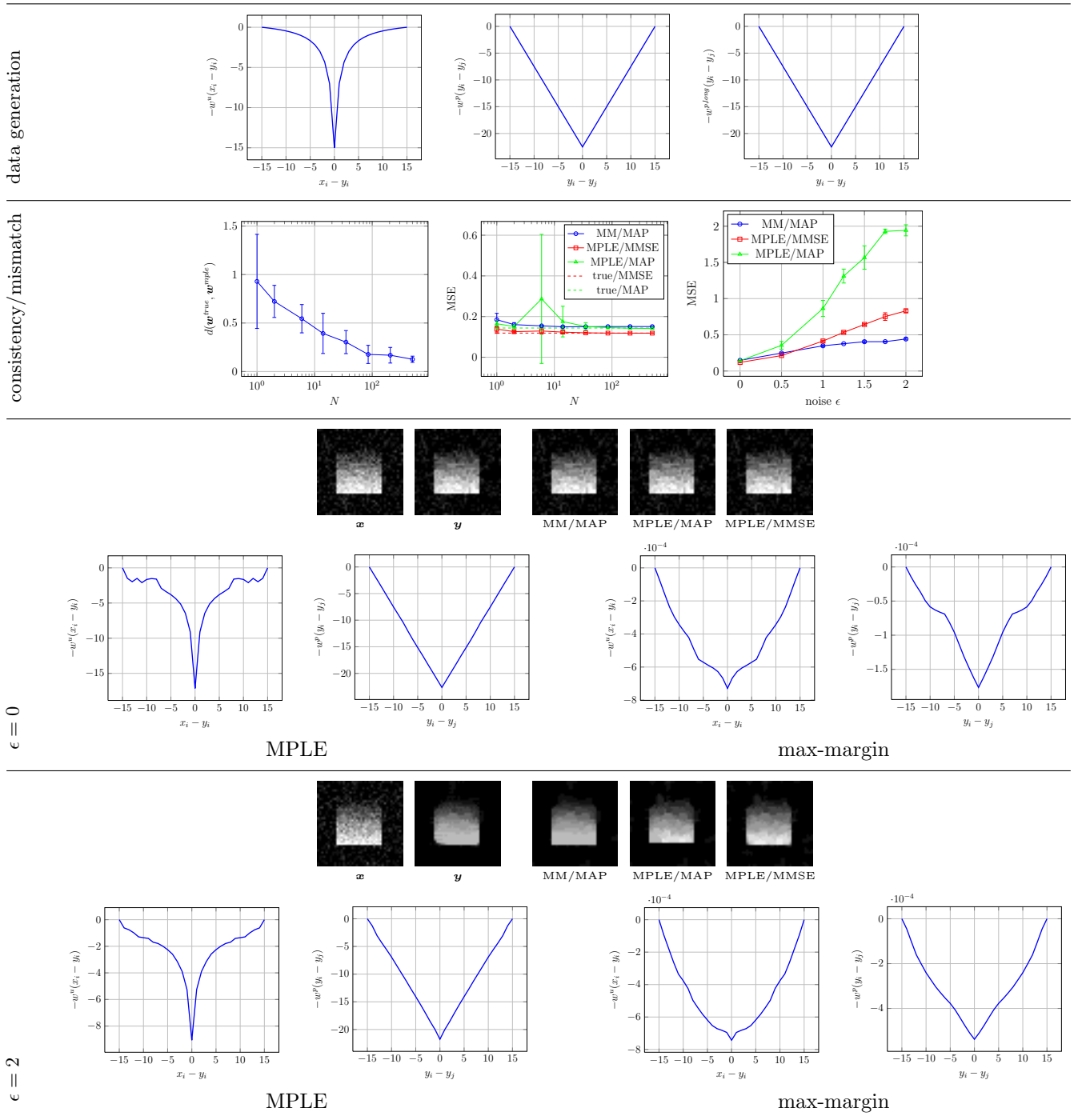


Fig. 3: First row: Weights used to generate the data: unary(left), pairwise - short distance (middle), pairwise - long distance (right). Second row: consistency of the MPLE (left); MSE result without misspecification, i.e. $\epsilon = 0$, (middle); and results for increased level of misspecification (right). Third row: Results for the case of no misspecification ($\epsilon = 0$). Last row: Results for the case of misspecification ($\epsilon = 2$).

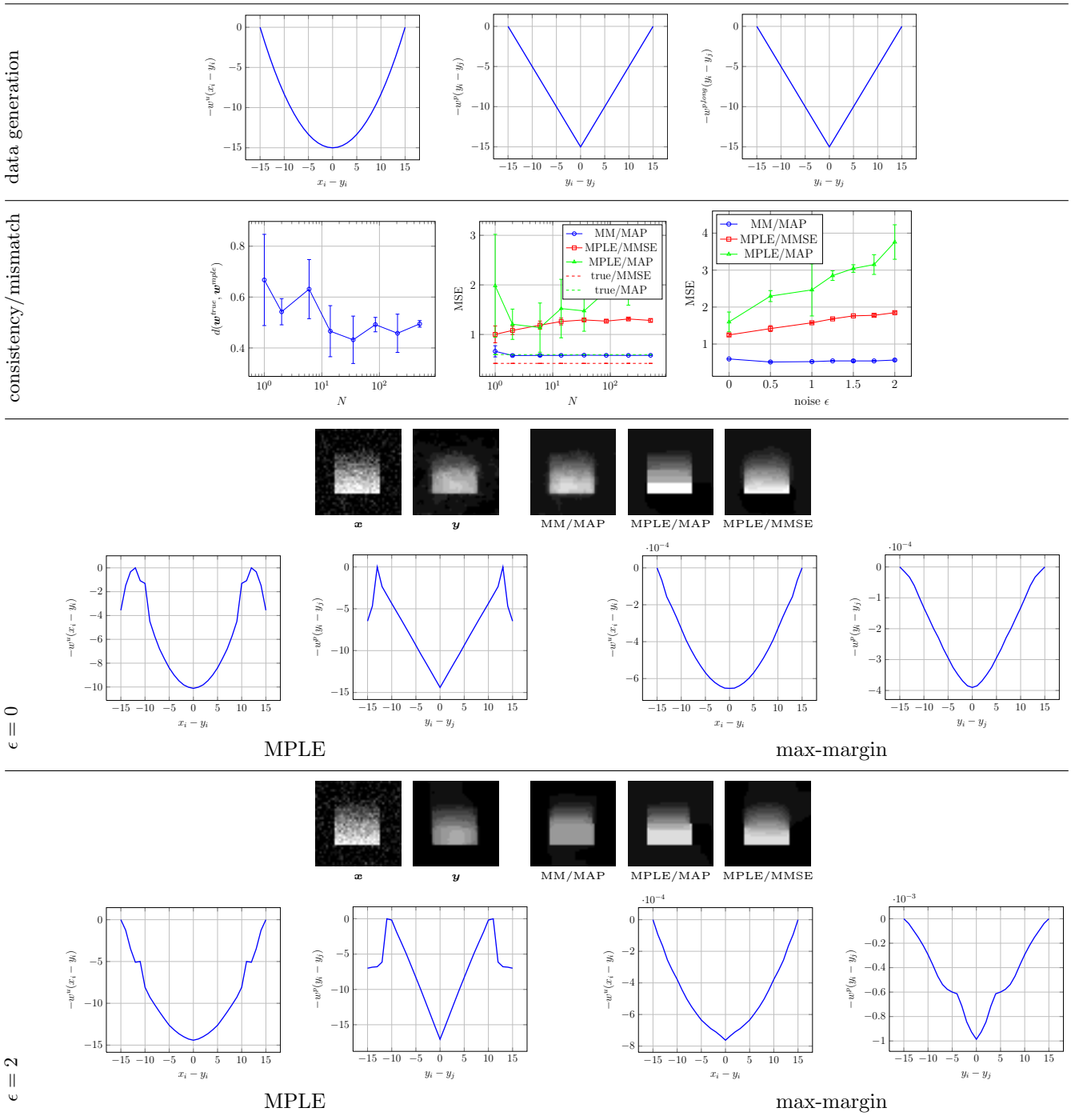


Fig. 4: First row: Weights used to generate the data: unary(left), pairwise - short distance (middle), pairwise - long distance (right). Second row: consistency of the MPLE (left); MSE result without misspecification, i.e. $\epsilon = 0$, (middle); and results for increased level of misspecification (right). Third row: Results for the case of no misspecification ($\epsilon = 0$). Last row: Results for the case of misspecification ($\epsilon = 2$).

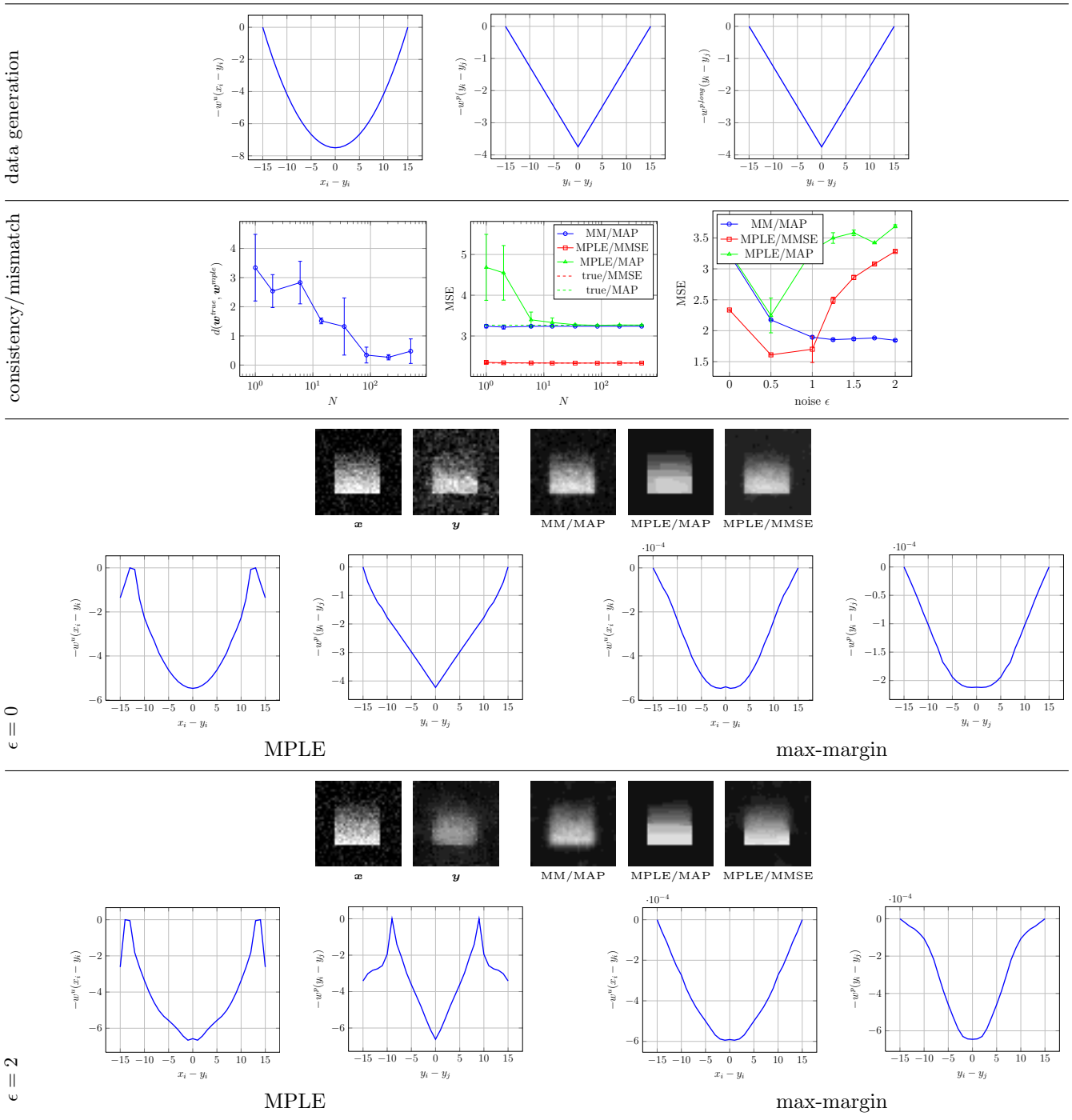


Fig. 5: First row: Weights used to generate the data: unary(left), pairwise - short distance (middle), pairwise - long distance (right). Second row: consistency of the MPLE (left); MSE result without misspecification, i.e. $\epsilon = 0$, (middle); and results for increased level of misspecification (right). Third row: Results for the case of no misspecification ($\epsilon = 0$). Last row: Results for the case of misspecification ($\epsilon = 2$).

4 Comment on PSNR error for Image denoising

There exist two definitions of the PSNR. The first is related to the mean-squared-error (MSE) and given by

$$PSNR(\boldsymbol{y}', \boldsymbol{y}) = 20 \log \left(\frac{K-1}{\sqrt{\frac{1}{\mathcal{V}} \sum_{i \in \mathcal{V}} (y_i - y_i')^2}} \right).$$

The second definition uses the standard deviation, which matches the MSE only if the mean error is equal to zero.

$$PSNR(\boldsymbol{y}', \boldsymbol{y}) = 20 \log \left(\frac{K-1}{\sigma} \right).$$

with σ the standard deviation of the pixelwise error. We use the second definition as it is more common in the image denoising literature. However, the difference between the two PSNR definitions is in practice rather small.

5 Additional results for Image denoising

Fig. 6 shows the learned weights for maximum margin and maximum pseudo-likelihood when BM3D is used as a second feature. Concerning Max-margin training (top row), as expected the unary weights of the BM3D output are higher than the ones of the noisy input. The pairwise terms (top row, right image) show that far less smoothing is needed in contrast to the case where BM3D is not available as feature (Fig. 4 in submitted paper). Fig. 7 shows the unary and pairwise image statistics of the predictions without BM3D (pairwise is the same as in the paper). Fig. 8 shows the image statistics of the predictions when BM3D is used. Fig. 10 shows extended results for the capricorn image used in the paper. Finally, at the end of the supplementary material we show some more predictions for other images. We decided to choose the most informative results: MM/MAP; MPLE/MMSE; BM3D; and MM/MAP with BM3D. For all dataset the results of BM3D and MM/MAP with BM3D are visually best. Although, MM/MAP with BM3D is always marginally better than BM3D in terms of PSNR, it is hard to argue that this difference is also visible in the results. Other approaches which use BM3D as additional feature are inferior, both visually and in terms of PSNR.

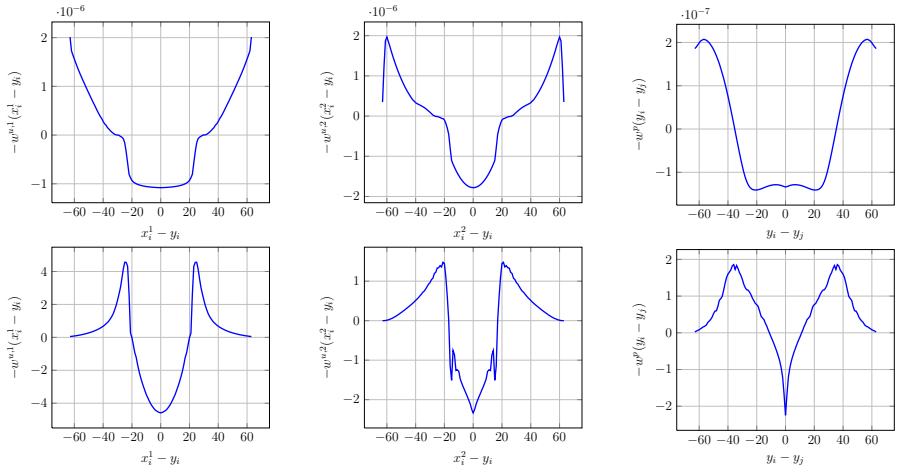


Fig. 6: Learned potentials when BM3D prediction is integrated as a second feature. First row: weights learned by maximum margin. Second row: weights learned by maximum pseudo-likelihood. From left to right: weights corresponding to the noisy image, weights corresponding to the BM3D output, and pairwise weights.

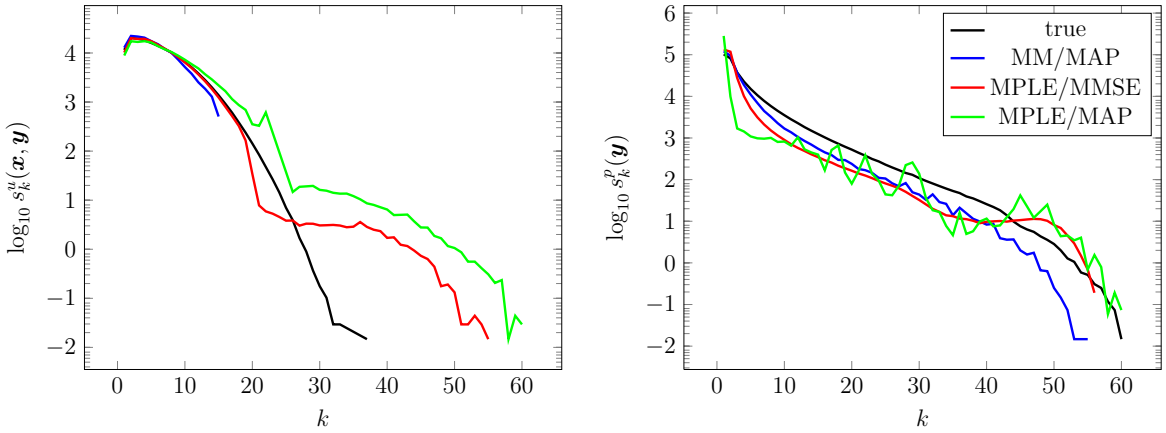


Fig. 7: Aggregated statistics on the test set.

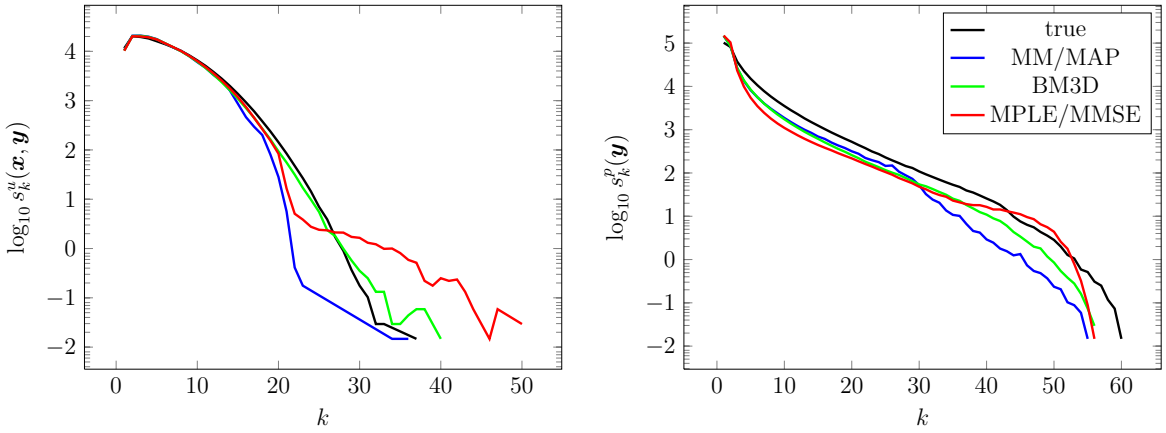


Fig. 8: Aggregated statistics on the test set when BM3D is used as an additional feature.

(a) noisy image, 20.29dB (b) original image (c) MM/MAP, 26.03dB

(d) MPLE/MMSE, 25.44dB (e) MPLE/MAP, 23.11dB (f) BM3D, 26.19dB

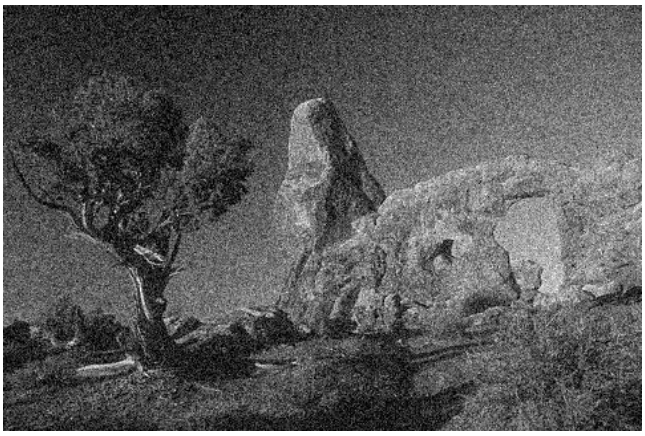
(g) MM/MAP with BM3D, 26.27dB (h) MPLE/MMSE with BM3D, 25.82dB (i) MPLE/MAP with BM3D, 23.95dB

Fig. 9: Results for the cropped capricorn image. (a) and (b) show the noisy and original image. (c)-(e) show predictions with the standard unary and pairwise costs. (g)-(i) show predictions with the BM3D prediction as an additional unary feature. (f) shows the result of the BM3D algorithm.

(a) noisy image, 20.29dB (b) original image (c) MM/MAP, 26.03dB

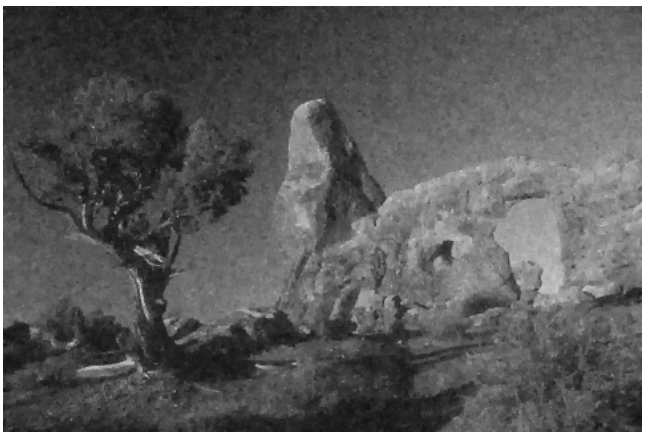
(d) MPLE/MMSE, 25.44dB (e) BM3D, 26.19dB (f) MM/MAP with BM3D, 26.27dB

Fig. 10: Results for the full capricorn image. (a) and (b) show the noisy and original image. (c) and (d) show predictions with the standard unary and pairwise costs. (e) shows the result of the BM3D algorithm. (f) shows the results of MM/MAP with BM3D as an additional feature.



(a) noisy image, 20.28dB

(b) original image



(c) MM/MAP, 27.11dB

(d) MPLE/MMSE, 26.65dB

(e) BM3D, 27.46dB

(f) MM/MAP with BM3D, 27.58dB

Fig. 11: The arch image. MM/MAP is better than MPLE/MMSE at reconstructing the fine structures. However, it is worse for the sky.

(a) noisy image, 20.82dB

(b) original image

(c) MM/MAP, 26.95dB

(d) MPLE/MMSE, 26.30dB

(e) BM3D, 27.88dB

(f) MM/MAP with BM3D, 27.91dB

Fig. 12: The foxes image. MM/MAP reproduces the fur better.

(a) noisy image, 20.21dB

(b) original image

(c) MM/MAP, 28.64dB

(d) MPLE/MMSE, 28.98dB

(e) BM3D, 29.69dB

(f) MM/MAP with BM3D, 29.74dB

Fig. 13: The elephant image. MPLE/MMSE is better at reconstructing the sky, MM/MAP better for the grass. Overall, MMSE leads to a better prediction.

(a) noisy image, 20.49dB

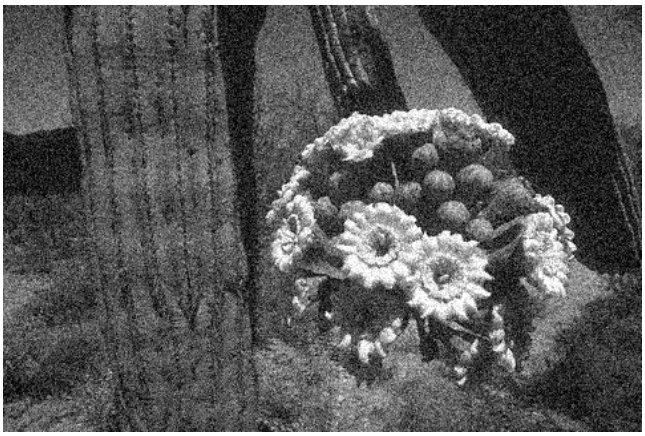
(b) original image

(c) MM/MAP, 24.79dB

(d) MPLE/MMSE, 23.85dB

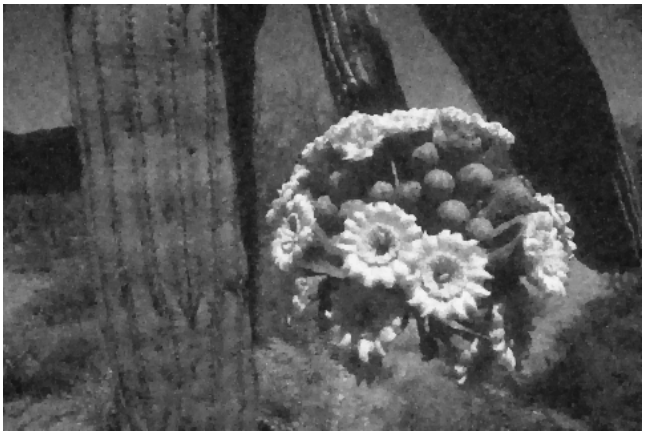
(e) BM3D, 25.14dB

(f) MM/MAP with BM3D, 25.24dB

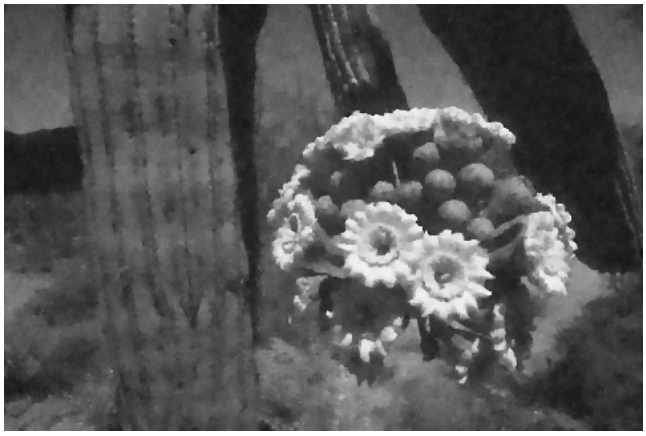


(a) noisy image, 20.39dB

(b) original image



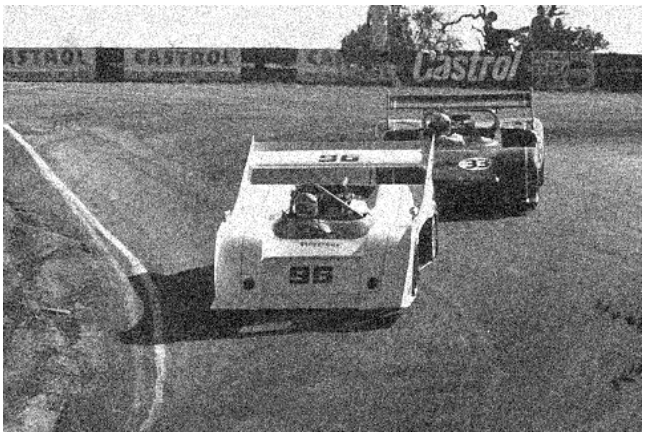
(c) MM/MAP, 26.85dB



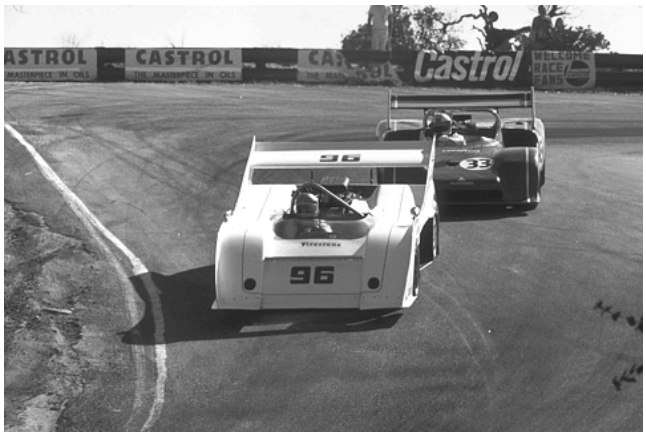
(d) MPLE/MMSE, 26.38dB

(e) BM3D, 27.31dB

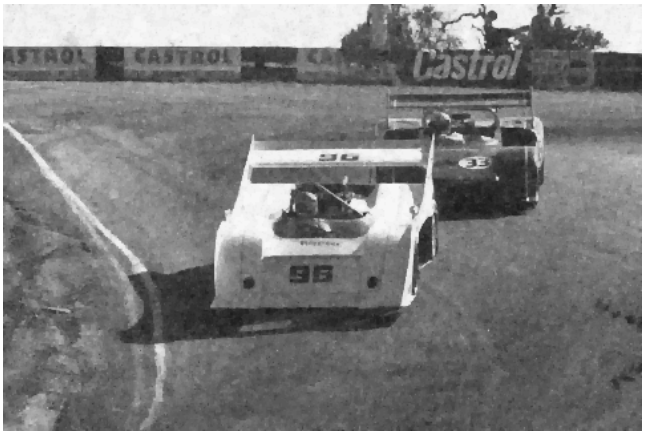
(f) MM/MAP with BM3D, 27.41dB



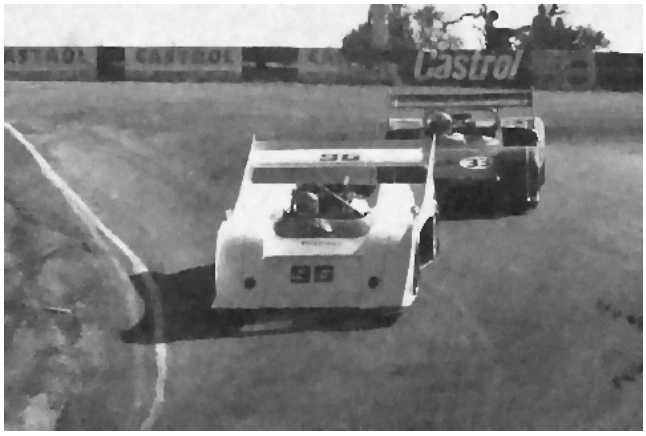
(a) noisy image, 20.41dB



(b) original image

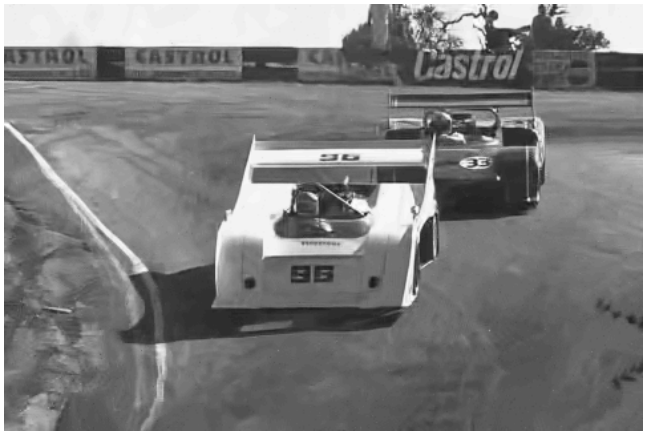


(c) MM/MAP, 27.22dB

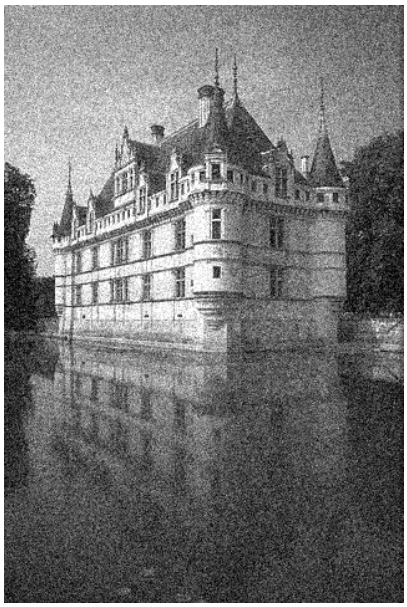


(d) MPLE/MMSE, 26.83dB

(e) BM3D, 28.43dB

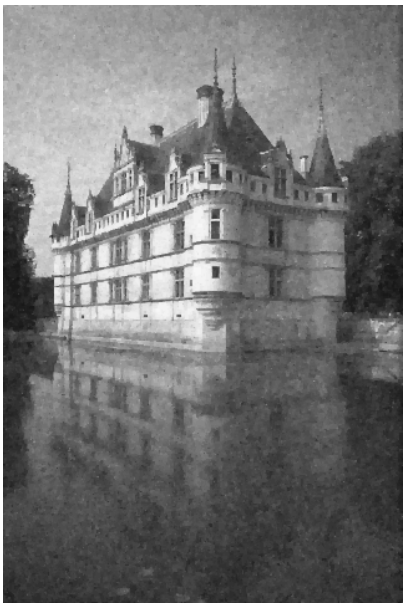


(f) MM/MAP with BM3D, 28.45dB

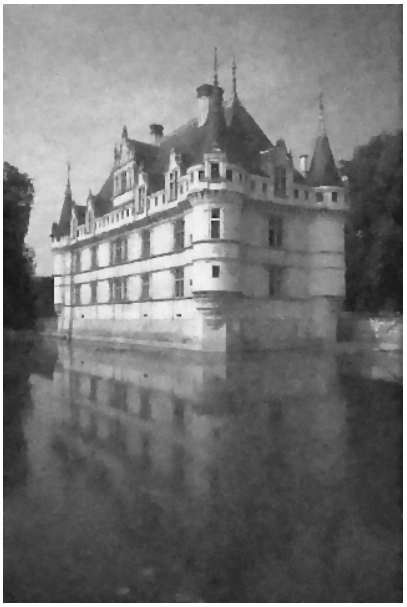


(a) noisy image, 20.28dB

(b) original image



(c) MM/MAP, 27.43dB



(d) MPLE/MMSE, 27.10dB

(e) BM3D, 29.23dB

(f) MM/MAP with BM3D, 29.25dB