Czech Pattern Recognition Workshop 2000, Tomáš Svoboda (Ed.)
Peršlák, Czech Republic, February 2–4, 2000
Czech Pattern Recognition Society

Robust Spot Fitting for Genetic Spot Array Images

Horng-Yang Chen, Norbert Brändle, Horst Bischof
Vienna University of Technology
Pattern Recognition and Image Processing Group
Favoritenstr. 9/1832, A-1040 Vienna
nob@prip.tuwien.ac.at

Hilmar Lapp
Novartis Research Institute
Genetics
Brunner Str. 59, A-1235 Vienna
hilmar.lapp@pharma.novartis.com

Abstract

In this paper we address the problem of reliably fitting parametric and semi-parametric models to high density spot array images obtained in genetic expression experiments. The goal is to measure the amount of genetic material at specific spot locations. A lot of spots can be modelled accurately by a Gaussian shape. In order to deal with highly overlapping spots we use robust M-estimators. When the parametric method fails (which can be detected automatically) we use a novel, robust semi-parametric method which can handle spots of different shapes accurately. These techniques are evaluated in experiments.

1 Introduction

Genetic spot array images have to be analyzed in the course of high-throughput hybridization experiments [2, 1], where a spot in the array can identitify specific expressed gene products. Biological systems read, store and modify genetic information by molecular recognition. Because each DNA strand carries with it the capacity to recognize a uniquely complementary sequence through base pairing (A↔T, C↔G) [5], the process of recognition, or *hybridization*, is highly parallel, as every nucleotide in a large sequence can in principle be queried at the same time. Thus, hybridization can be used to efficiently analyze large amounts of nucleotide sequence. The primary approaches include array-based technologies that can identify specific expressed gene products on high density formats, including filters, microscope slides and microchips [2]. Common to all array-based approaches is the necessity to analyze digital images of the array. The ultimate image analysis goal is to automatically assign a quantity to every array element giving information about the hybridization signal (*spot fitting*). Figures 1 and 2 show a typical array image generated in the course of a oligonucleotide fingerprint (ONF) experiment: The high density medium is a filter (nylon membrane) comprising a total of 57600 cDNA [5] spots which were spotted in different spotting cycles by a robot arm carrying a matrix of needles. Detailed information about the spotting procedure can be found in [7]. The intensity of every spot corresponds to the amount of label remaining after hybridizing a liquid containing the labelled probes and subsequently washing off probe not bound to the genetic material. For details about the physical imaging process refer to [4]. The *grid fitting* provides coarse inital locations of the spots and is described in detail in [1]. The goal of spot fitting is to provide an accurate estimate of the volume, i.e. the amount of genetic material of every spot. It must cope with the following three major problems:

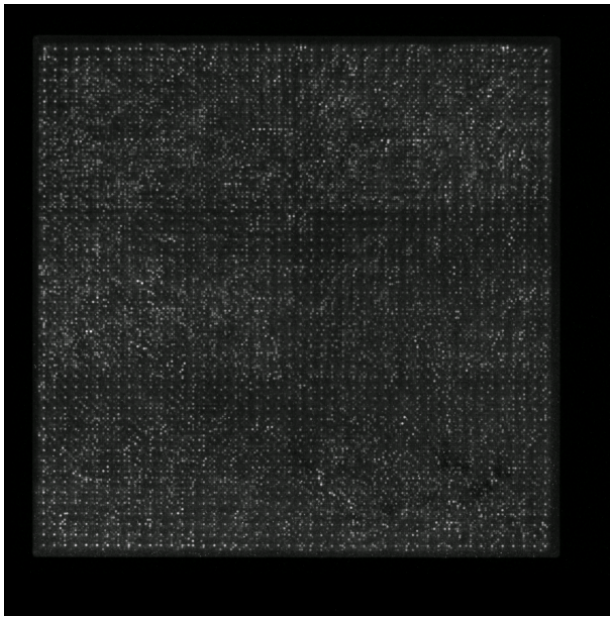


Figure 1: Genetic Spot Array Image. The white rectangle indicates the region belonging to the zoomed right image shown in Fig. 2

1. *Noise and outliers*: Sometimes gross errors like artifacts which do not comprise gene expression information can occur (Fig. 3a).

2. *Overlapping spots*: Spots with high intensity may interfere with neighboring spots (Fig. 3b).

3. *Various spot shapes*: Depending on the type of the experiment different spot shapes are possible. We therefore cannot always assume a parametric spot model (Fig. 4).

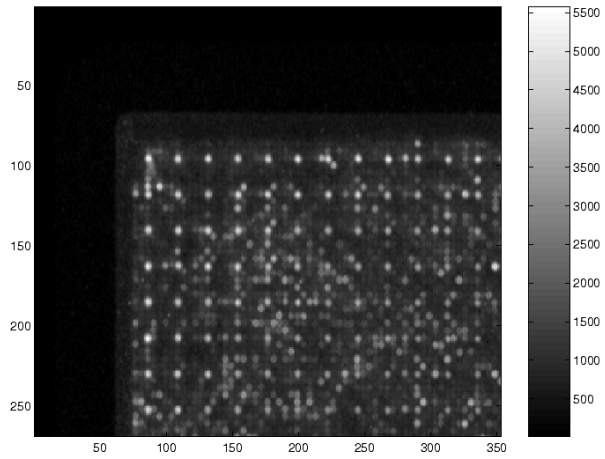
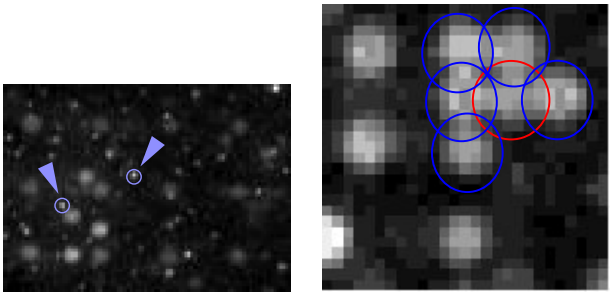


Figure 2: Genetic Spot Array Image. Zoomed image corresponding to the white rectangel of Fig. 1



(a) Artifacts (b) Overlapping spots

Figure 3: Spot fitting problems

Noordmans and Smeulders [8] provided a general approach for the detection and characterization of overlapping spots. This approach is restricted to parametric models and is non-robust. In this paper we desribe both a parametric and a non-parametric approach for spot fitting.

2 Parametric Spot Fitting

The grid fitting provides us with approximate locations for every spot in the image. With this information we can assign a set of image (pixel) intensities to each spot using the prior size of a spot. On this local set of points around each spot one can perform a parametric fit. A parametric fit assumes a given analytic model where its unknown parameters are to be determined. In either case the procedures have to be robust. Since the initial spot locations provided by the grid fitting are quite accurate we can use M-estimators [6] as robust fitting procedures.

2.1 The Gaussian spot model

Let $S = \{(p_i, z_i),\ p_i \in \mathbb{R}^2,\ z_i \in \mathbb{R}\}$ be a set of points corresponding to a spot where z_i denotes the intensity at location p_i. An initial analysis has shown that most of the spots can

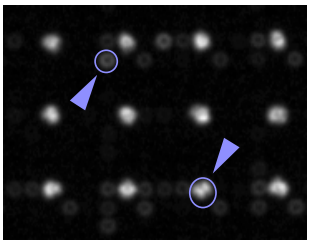


Figure 4: Spots with uncommon shapes

be characterized fairly accurately by a Gaussian shape. The Gaussian function is denoted as

$$G(p, \mu, \Sigma) = e^{-\frac{1}{2}(p-\mu)'\Sigma^{-1}(p-\mu)} \tag{1}$$

with $\mu \in \mathbb{R}^2$ and $\Sigma \in \mathbb{R}^{2\times 2}$ (2×2 matrix). Our Gaussian spot model is

$$Z(p, A, B, \mu, \Sigma) := A\, G(p, \mu, \Sigma) + B \tag{2}$$

with following parameters:

1. A is the amplitude of the Gaussian model corresponding to the "height" of the spot.
2. μ is the mean of the Gaussian model corresponding to the "center" of the spot.
3. Σ is the 2×2 dispersion matrix:
$$\Sigma = \begin{pmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{xy} & \sigma_y^2 \end{pmatrix}. \tag{3}$$
4. B is the background value. The background has to be modeled due to unspecific radioactivity. Since it is generally nonuniform across the spot image, we have to estimate B for every spot with a hierarchical approach using Gaussian image pyramids.

2.2 Relative error and Goodness of fit

In order to quantitatively assess how well the (Gaussian) model assumption holds we have to introduce a measure for the error. We apply an approach also used in linear regression analysis as in [3] by comparing the model to a "standard model Z_0":

$$T_1 := \frac{\frac{1}{n}\sum_{i=1}^{n}(z_i - Z(p_i))^2}{\frac{1}{n}\sum_{i=1}^{n}(z_i - z_0)^2} \tag{4}$$

where $z_0 := \frac{1}{n}\sum z_i$, which is the mean of the given values. The standard model Z_0 in this case is a plane parallel to the p_i-plane at the height z_0, i.e. $Z_0 \equiv z_0$. The interpretation of T_1 is, the fit of model Z is $1/\sqrt{T_1}$ times better than the standard model. We will call T_1 the *relative squared error* or for short *relative error*. In literature $1 - T_1$ is called the *goodness of fit*.

2.3 Non-robust parameter estimation

The parameters can be computed by maximum likelihood estimators and least square minimization.

2.3.1 Estimating the mean and the dispersion matrix

The parameter estimation of the Gauss model is based upon the following principle: The shape of a spot can be interpreted as a distribution of the x- and y-coordinates. A spot patch S containins $n = M_S \cdot N_S$ points p_i and has the intensities $I(p_i)$. Let us denote the corrected intensities as $z_i := \max(I(p_i) - \hat{B}, 0)$. Since the background can be overestimated especially in the first background estimation, we correct negative values to zero. The estimate $\hat{\mu}$ of the center μ and the estimate $\hat{\Sigma}$ of the dispersion matrix Σ are then computed by the maximum likelihood (ML) estimators:

$$\hat{\mu} = \frac{1}{T} \sum_{i=1}^{n} z_i p_i \tag{5}$$

and

$$\hat{\Sigma} = \frac{1}{T} \sum_{i=1}^{n} z_i (p_i - \hat{\mu})(p_i - \hat{\mu})' \tag{6}$$

where T is the total sum of the intensities of the patch:

$$T = \sum_{i=1}^{n} z_i. \tag{7}$$

Hence the estimate $\hat{\mu}$ of the center μ is given by the sample average (i.e. the average with respect to the given data set) of the coordinates weighted by the pixel intensities. Similarly, the maximum likelihood estimate $\hat{\Sigma}$ of the dispersion matrix Σ is given by the sample average of the outer product $(p_i - \hat{\mu})(p_i - \hat{\mu})'$ weighted by the pixel intensities.

2.3.2 Estimating the amplitude

The estimate $\hat{A}$ of the amplitude A is computed by a least squares minimization. Let us define the error function

$$E(A) = \frac{1}{2} \sum_{i=1}^{n} \left\{ A\, G(p_i, \hat{\mu}, \hat{\Sigma}) - z_i \right\}^2. \tag{8}$$

The estimate for A is computed by setting the partial derivative of E with respect to the parameter A to zero:

$$\frac{\partial E}{\partial A} = \sum_{i=1}^{n} \left\{ A\, G(p_i, \hat{\mu}, \hat{\Sigma}) - z_i \right\} G(p_i, \hat{\mu}, \hat{\Sigma}) = 0. \tag{9}$$

The solution to this equation yields the estimator $\hat{A}$:

$$\hat{A} = \sum_{i=1}^{n} z_i G(p_i, \hat{\mu}, \hat{\Sigma}) \Big/ \sum_{i=1}^{n} G(p_i, \hat{\mu}, \hat{\Sigma})^2. \tag{10}$$

2.3.3 Quantification

The brightness V of the spot is estimated as the volume under the fitted Gaussian function:

$$\hat{V} = \hat{A}\, (2\pi) \sqrt{|\hat{\Sigma}|}. \tag{11}$$

Since the scanner is square rooting the intensities during the scanning process we provide an estimator for the brightness W of the spot with squared intensities. Using the fact that $G(p, \mu, \Sigma)^2 = G(p, \mu, 2 \cdot \Sigma)$, one can easily verify that

$$\hat{W} = \hat{A}^2\, \pi \sqrt{|\hat{\Sigma}|}. \tag{12}$$

2.4 Robust parameter estimation

The non-robust estimators of Sect. 2.3 can be taken as starting points for the computation of robust estimators. The theory of robust M-estimators for multivariate distributions with elliptically symmetric density function is studied by Maronna [6]. We adapt these results to suit to our needs and introduce a joint estimation for the mean, the dispersion matrix and amplitude.

2.4.1 Estimating the mean

$$\tilde{\mu} = \sum_{i=1}^{n} w_1(e_i) z_i p_i \Big/ \sum_{i=1}^{n} w_1(e_i) z_i \tag{13}$$

where

$$w_1(x) = \frac{\psi(x)}{x} \tag{14}$$

with Tukey's biweight as the ψ - function:

$$\psi(x) = \begin{cases} x \left(1 - \left(\frac{x}{a}\right)^2\right)^2 & , \quad |x| \le a \\ 0 & , \quad |x| > a \end{cases} \tag{15}$$

e.g. $a = 4$ and e_i as the studentized error for each point

$$e_i := e_i(\tilde{A}, \hat{B}, \tilde{\mu}, \tilde{\Sigma}) := (\tilde{A}\, G(p_i, \tilde{\mu}, \tilde{\Sigma}) - z_i)/\sigma \tag{16}$$

with unknown spread σ.

2.4.2 Estimating the dispersion matrix

$$\tilde{\Sigma} = \frac{1}{T} \sum_{i=1}^{n} w_1(e_i)^2 z_i (p_i - \tilde{\mu})(p_i - \tilde{\mu})' \tag{17}$$

with T as the total sum of the intensities of the patch (Eq. 7).

2.4.3 Estimating the amplitude

$$\tilde{A} = \frac{\sum_{i=1}^{n} w_1(e_i) z_i G(p_i, \tilde{\mu}, \tilde{\Sigma})}{\sum_{i=1}^{n} w_1(e_i) G(p_i, \tilde{\mu}, \tilde{\Sigma})^2}. \tag{18}$$

This amplitude estimation is the result of an error-minimization problem with the help of a ρ-function [6] as follows:

$$R(A) == \sum_{i=1}^{n} \rho(e_i) \;\rightarrow\; \min \tag{19}$$

will yield a scale invariant M-estimator $\tilde{A}$ for A. Differentiating equation 19 with respect to $\tilde{A}$ results in

$$\sum_{i=1}^{n} \psi \left(\frac{A\, G(p_i, \tilde{\mu}, \tilde{\Sigma}) - z_i}{\sigma} \right) \frac{\partial \left(A\, G(p_i, \tilde{\mu}, \tilde{\Sigma}) - z_i \right)}{\partial A} = 0. \tag{20}$$

2.4.4 Parameter computation The equations for $\tilde{\mu}$ (13), $\tilde{\Sigma}$ (17) and $\tilde{A}$ (18) can be solved by the weighted least square iteration:

$$\mu_{j+1} = \sum_{i=1}^{n} w_1(e_{ij}) z_i p_i \Big/ \sum_{i=1}^{n} w_1(e_{ij}) z_i \tag{21}$$

$$\Sigma_{j+1} = \frac{1}{T} \sum_{i=1}^{n} w_1(e_{ij})^2 z_i (p_i - \mu_j)(p_i - \mu_j)' \tag{22}$$

$$A_{j+1} = \frac{\sum_{i=1}^{n} w_1(e_{ij}) z_i G(p_i, \mu_j, \Sigma_j)}{\sum_{i=1}^{n} w_1(e_{ij}) G(p_i, \mu_j, \Sigma_j)^2} \tag{23}$$

with

$$e_{ij} = (A_j\, G(p_i, \mu_j, \Sigma_j) - z_i)/\sigma_j \tag{24}$$

and

$$\sigma_j = \text{median}_{i \in I^*} \frac{|A_j\, G(p_i, \mu_j, \Sigma_j) - z_i|}{0.6745}. \tag{25}$$

where $I^* = \{i | G(p_i, \mu_j, \Sigma_j) < c\}$, e.g. $c = 1.6 * u_{0.95}$ where $u_{0.95}$ denotes the 0.95 quantile of the standard normal distribution.

2.5 Managing overlapping spots

In order to save resources the genetic material is spotted with high density with the consequence that spots may overlap. Especially spots with high intensity may interfere with neighboring spots, see Fig. 3b. The problem when fitting a model is that it will be biased towards the overlapping neighbor yielding a dislocation of the fitted model and a too high quantification. One possible method is to correct the input intensities for a spot by subtracting overlapping neighboring models. However, usually too much is subtracted due to the overlapping situation, such that an iteration process between subtracting neighbor models and refit is needed. Another possible approach the usage of robust estimators. Intensities which are too high due to overlapping are regarded as outliers and are subsequently downweighted. In our paper we use a combination of the two schemes: In a first step a robust fit is performed, then the background estimation is improved and finally a robust refit is performed on the data with subtracted neighboring models.

Subtracting neighboring models Let $\mathcal{G} = \{g_{ij} | i \in \{1, \ldots, I_G\}, j \in \{1, \ldots, J_G\}\}$ be the set of spots. For each spot g_{ij} let us assume we have computed a model $Z_{ij}(p, q)$ with the parameters $q \in \mathbb{R}^k$ and coordinate $p \in \mathbb{R}^2$. Consider the $M_S \times N_S$ image patch $S_{ij} = S_{ij}(p)$ for g_{ij}. In order to take overlapping spots into account we can recompute the model Z_{ij} by using the modified spot patch

$$S^*_{ij} = S_{ij} - \sum_{k,l \in \{-1,0,1\}, (k,l) \neq (0,0)} Z_{i+k\ j+l} \tag{26}$$

i.e. subtracting neighboring spot models. Further we set the models $Z_{ij} := 0$ for $i \in \{0, I_G + 1\} \vee j \in \{0, J_G + 1\}$ to deal with the special cases of border points. One could iterate this procedure for every spot g_{ij} over the whole image. One then gradually obtains better models for every spot, stopping when the parameters of the model for each spot stabilize.

The approach with the robust estimators will find good fits without "iterating over the whole image" as done in the first method. In addition, spots with shapes which are not covered directly by a model as the 'volcano shape' in Fig. 4 can be treated because the robust estimator will react less sensitive to abnormities or gross errors like artifacts (see Fig. 3a).

3 Semi-parametric Spot Fitting

A semi-parametric approach can describe the spot shape more accurately in the case of deviations from the model assumptions, which is the case in Fig. 4. However, overlap handling will be difficult, because a semi parametric fit will lack an intrinsic declension of the tails of a parametric model.

3.1 Algorithm

The basic idea of this method is to reduce dimensionality of given data using prior knowledge. Assuming that the spot has elliptically symmetric shape the fit is computed in the following steps:

A. Find the spot center We first perform a Gaussian fit computing M-estimators for $\tilde{\mu}$ and $\tilde{\Sigma}$ as described in Sect. 2.4.1 and Sect. 2.4.2 . $\tilde{\mu}$ is our center. Since the M-estimator of the location is robust it will also deal with spots with uncommon shapes. Passing a line perpendicular to the x, y-plane through $\tilde{\mu}$ gives us the axis a.

B. Transform the points The estimated dispersion matrix $\tilde{\Sigma}$ gives us an ellipse in x, y-plane. Let e_1 and e_2 be the two eigenvalues of $\tilde{\Sigma}$, (without loss of generality $e_1 \geq e_2$), v_1 and v_2 the corresponding eigenvectors and ϵ be the half-plane spanned by $\lambda_1 a + \lambda_2 v_1;\ \lambda_1 \in \mathbb{R},\ \lambda_2 \in \mathbb{R}_0^+$. Consider the one parametric family of ellipses with the principle axis directions v_1 and v_2, and diameters λe_1 and λe_2, $\lambda \in \mathbb{R}_0^+$ and center μ. The family covers the x, y-plane without intersection, each point in the x, y-plane lies exactly on one ellipse. We "rotate" the given points (p_i, z_i) following the path corresponding to p_i into the half-plane ϵ yielding a point cloud q_i in 2-space (see Fig. 11f. The first coordinate can be easily computed by:

$$e_1 \cdot |p|^2 / \sqrt{e_1^2 (p \cdot v_1)^2 + e_2^2 (1 - (p \cdot v_1)^2)} \tag{27}$$

the second is the unchanged z-coordinate.

C. Compute a profile We introduce a simplified, efficient and robust version of curve approximation for scattered points suited to our purpose. First we compute m points $c_i = (x_i, y_i),\ i = 1, \ldots, m$ well describing the shape of curve to be computed. Consider the vertical parallel strip with y-axis and $x \equiv \max x_i$ as borders. We then segment the strip into m commensurate parallel strips and compute $c_i = \text{median}_k r_{ki}$, where r_{ki} are those points q_k lying in the

i^{th} strip, see Fig. 11f. We further cut off tails of the profile by gradually lowering the profile points down to zero in the last quarter, because 1) especially at the tail there may be some overlapping situation and 2) generally there are fewer points at the tail. For our purpose it is enough to interpolate the points c_i by a polygon and to perform asmoothing scheme on the profile points, e.g. by replacing each point with a weighted sum of its neighbors. Alternatively one can compute a spline interpolating the points c_i for the profile curve.

D. Compute Volume The profile curve is rotated following the elliptical paths as in step B.

3.2 Quantification

Let $c_i = (x_i, y_i),\ i = 1, \ldots, m$ be the profile points as introduced in the previous paragraph. The brightness V of the spot is then estimated by taking

$$\hat{V} = \frac{e_2}{e_1} \cdot \frac{1}{3} \sum_{i=2}^{m} (x_{i-1}^2 + x_{i-1}x_i + x_i^2)\pi(y_i - y_{i-1}). \quad (28)$$

The brightness W of the spot with squared intensity is

$$\hat{W} = \frac{e_2}{e_1} \cdot \frac{1}{3} \sum_{i=2}^{m} (x_{i-1}^2 + x_{i-1}x_i + x_i^2)\pi(y_i^2 - y_{i-1}^2). \quad (29)$$

If one desires good results one should use known numerical integration schemes as (composite) Simpson's rule.

4 Spot Detection Limit

A spot fitting algorithm should decide whether a location contains a spot before performing a fit. Imagine having input intensities with perfect zero values, computing the mean would lead to a division by zero or leading to a singular dispersion matrix. This can happen rather often since the first background estimation is overestimating the background.

One could use our test for goodness of fit as spot detection by testing the "Zeromodel" $Z_{zero} \equiv 0$ being 'd-appropriate' or not, using the test statistic:

$$T_1 := d^2 \cdot \frac{\sum_{i=1}^{n} z_i^2}{\sum_{i=1}^{n} (z_i - z_0)^2} \quad (30)$$

e.g. $d^2 = 2$. However, the results were not satisfying because the values did not well separate spot locations and non spot locations. We use instead

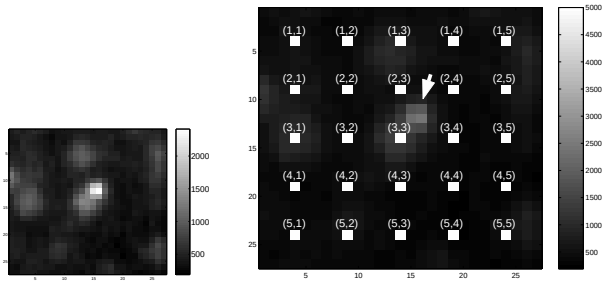
$$T_2 := \text{median} z_i > d \quad (31)$$

for spot detection where $d = log(2) \cdot V^*/(M_S \cdot N_S)$ and V^* is the minimum volume a location carries where a spot still can be expected. The interpretation is that if the volume of a location $V is greater than V^*$, we expect that there is a spot. The easiest way to estimate the volume is $M_S \cdot N_S \cdot \sum z_i$, leading to $T_{2a} := \sum z_i > V^*/(M_S \cdot N_S)$. To overcome noise and outliers for example due to overlaps we would like to use the median. Assuming that z_i is exponentially distributed which comes close to our situation, the $log(2) \approx 0.6931$ times the median estimates the mean. Replacing the mean by the median yields T_2.

5 Experimental Results

5.1 Artifacts

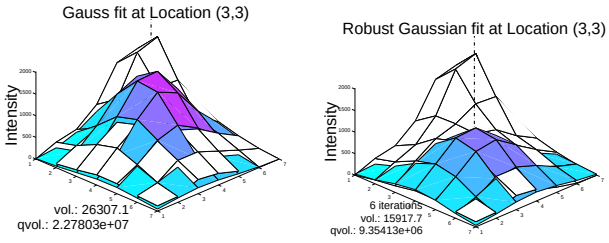
Consider the patch in Fig. 5a. The prior spot locations after the grid fitting are shown in Fig. 5b. The spot $(3, 3)$ in the



(a) Image data (b) Prior spot locations

Figure 5: Given patch with artifact

center is distorted by an artifact. As can be seen in Fig. 6a, a simple Gaussian fit will fail, because the location is biased towards the location of the artifact. The robust Gaussian fit can overcome the outlier. Figure 6b shows the result after 6



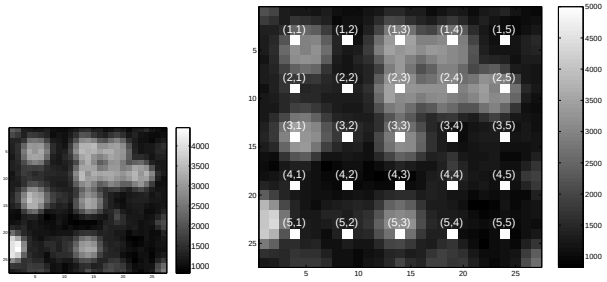
(a) (b)

Figure 6: Gaussian fit and robust Gaussian fit for spot $(3, 3)$

iterations. The label "vol." denotes the volume of the spot and "qvol." denotes the volume with squared intensities.

5.2 Overlapping Spots

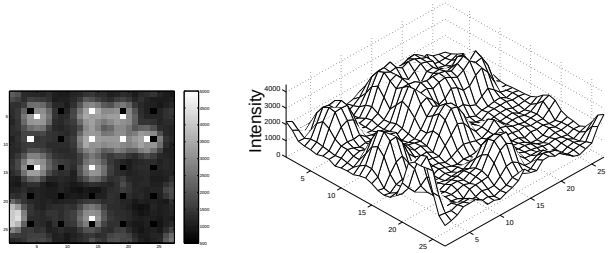
We demonstrate how the robust Gaussian fit works on image data with overlapping spots. Figure 7a shows a 5×5 block originating from an ONF image with low resolution. Figure 7b shows the prior spot locations after the grid fitting. Before a fit is performed a spot detection limit as introduced in Eqn. 31 is computed with limit $V^* = 30000$ corresponding to $d = 400$. In Fig. 8a the white marks indicate the detected spots. Fig. 8b shows a 3D plot of the block. Almost



(a) Image data (b) Detected spots

Figure 7: Block with overlapping spots

all locations are classified correctly including location $(5,1)$, where a neighboring spot is interfering from the left. Location $(2,1)$ is falsly detected as a spot, because two neighboring spots are overlapping. Spot (3,1) is an ordinary spot with no interfering neighbors, the robust estimator stops after 3 iterations without any big changes. Spots $(1,3)$, $(2,5)$



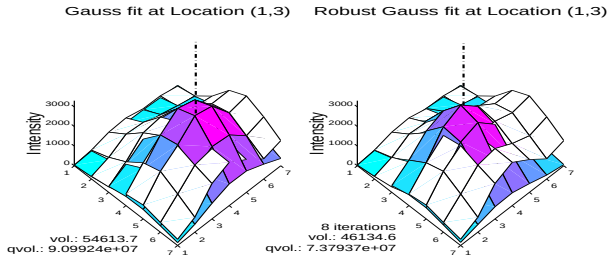
(a) Detected spots (b) 3D plot

Figure 8: Detected spots and 3D plot of image data

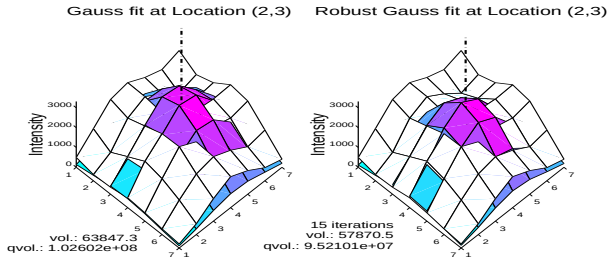
and $(3,3)$ have up to three overlapping neighbors, here the robust estimator can recover the original spot location quite well, especially for $(1,3)$ and $(3,3)$. Spot $(1,3)$ is plotted in Fig. 9a. The non-robust Gaussian fit is biased towards the neighboring spots, whereas the location of robustly fitted Gaussian spot is more plausible. Spots $(1,4)$, $(2,3)$ and $(2,4)$ have over four overlapping neighbors and are therefore difficult cases, but still some improvements can be done. The non-robust and robust Gaussian fits are plotted in Fig. 9b.

After the first robust Gauss fit we refit on every location with subtracted neighborhood models. The centers computed during the first fit are taken as the a priori centers for the second fit. When taking a look at the new patches with subtracted neighbors (see Fig. 10a) one will notice that the patches are now less distorted than the previous patch and are more "spot like" – an indication that the situation has improved.

When investigating the goodness of fit and the patch shapes, the first robust fitting resolved the overlaps at spos $(1,3)$ (see Fig. 10a) and $(3,3)$ very well. The results for



(a) Initial spot fitting for spot (1,3)



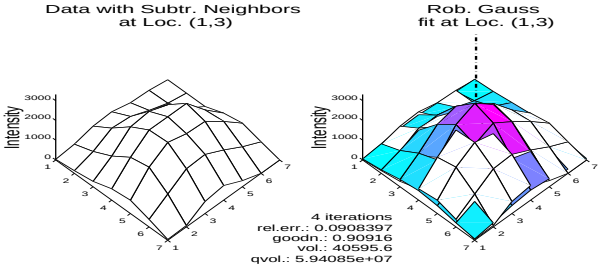
(b) Initial spot fitting for spot (2,3)

Figure 9: Initial non-robust and robust Gaussian spot fitting

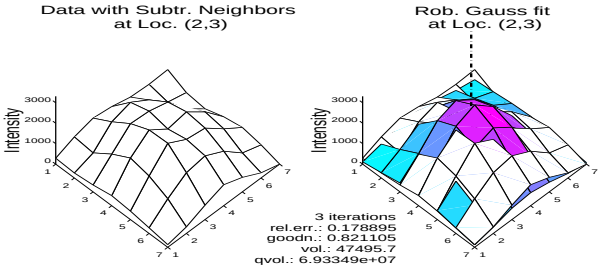
the spots (1,4) and (2,5)are good, the results for (2,3) (see Fig. 10b) are acceptable, and the results for (2,4) are not good enough. Generally, on can say that the robust estimation will perform well up to four overlapping neighbors while more than four will make problems. This is can be explained by the fact that highest possible breakdown point of a robust estimator is $\epsilon^* = 0.5$. If more than 50% of the input datas are false the situation cannot be recovered directly by a robust estimator. An overview of the fitted models can be seen in Fig. 10.

5.3 Uncommon Shapes

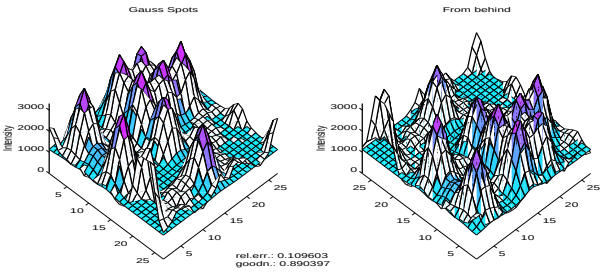
Figures 11a and b show a volcano spot with an overlap from the right hand side. An ordinary Gaussian fit would be biased to the right neighbor, but a robust estimator recovers the location easily (Fig. 11c). Performing a robust Gauss fit on both sides we subtract the neighborhood spot model from the patch receiving the corrected data (see Fig. 11d). After a Gaussian refit the initial volume estimation can be observed in Fig. 11e but the estimated volume is not very reliable due to the high relative error rate. Using the center and dispersion we performed a semi parametric fit (see Fig. 11f). We smoothed the profile points by replacing each point (except at the border) with the weighted sum over the left, the point itself and right neighbor with the weights 3,6, and 2. The left neighbor received higher weights, because the points on the left hand side are more reliable since they are closer to the center. The goodness of fit improved and a more reliable quantification is done. We also compared the algorithms to each other by plotting the percentage of data covered by the strip with the two offset profile curves as borders yielding a



(a) After subtraction of neighborhood models of spot (1,3)



(b) After subtraction of neighborhood models of spot (2,3)



(c) Models of all detected spots

Figure 10: Detected spots and 3D plot of image data

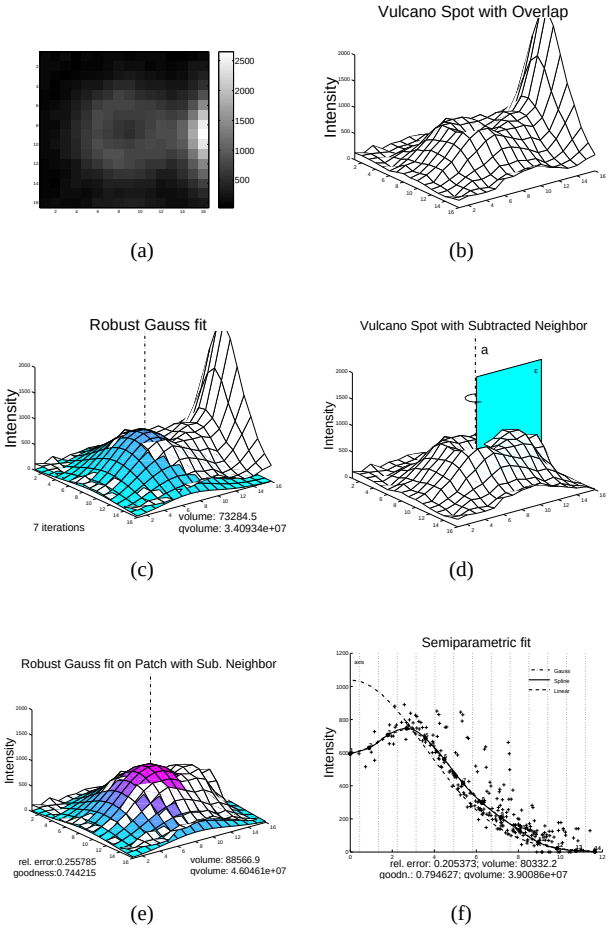


Figure 11: Volcano spot with overlapping neighbor

performance curve, see Fig. 12. A quick ascending curve indicates that the method is performing well, because the data points are covered early. As one can see the semi-parametric fit is better than the Gaussian fit.

5.4 Entire Image

We demonstrate the result of the spot fitting for the image in Fig. 2 containg a total of $40 \times 60 = 240$ spots. After the grid fitting we have the prior locations of every spot and apply the first background estimation routine to be ready for the first run. A major issue is the detection limit. It determines whether the location possibly contains a spot of interest. Pretending we do not know much about the volume of a spot we set $V^* = 0$ using the detection limit Eqn. 31. The algorithm will then fit at every location. After a second backgound estimation and a second run the fits from the first run are refined. The reconstructed image can be seen in Fig. 13. An analysis showed that most of the spots have a volume between 10000 and 20000 – locations with a volume lower than 10000 possibly contain no spots. With this knowledge we set V^* to 10000 and rerun our program. This time no singular locations are detected. In Fig. 14 we plotted the relative error for each spot location. Some locations have a significant error over 1.0. This is due to a bad fit as the result of fitting a model to location containing no spot. A location without a spot can still pass the detection limit when neighboring spots are interfering. We therefore perform a post processing procedure by simply rejecting a model with too high error, e.g. relative error greater that 0.4. In general, a relative error smaller than 0.1 indicates a very good, smaller than 0.2 a good fit, while at values bigger than 0.4 or 0.5 the fit should be rejected.

5.5 Complexity

Table 1 shows the CPU-time costs for each method per fit in flops. The values should be interpreted as follows:

Resolution →	Low Res. 7x7	High Res. 16x16
Method ↓	flops/per fit	flops/per fit
Gauss fit	10.000	47.000
Semi parametric fit	2.000	15.000

Table 1: CPU-time in flops, Re. = Resolution

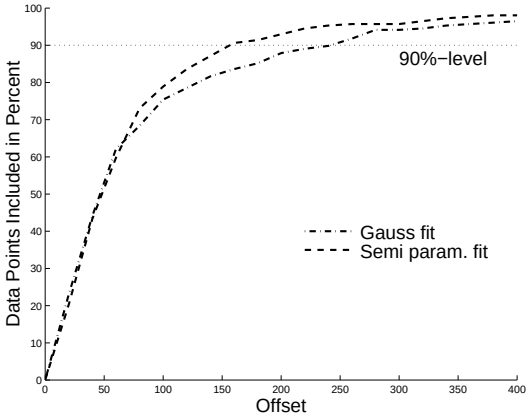


Figure 12: Semi-parametric fit

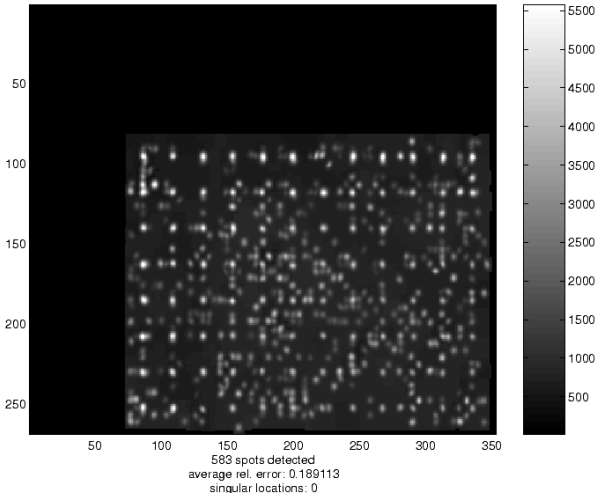


Figure 13: Reconstructed spot image of the spot image in Fig.2

1. A (non robust) Gaussian fit in low resolution requires approximately 10.000 flops.

2. A robust Gaussian fit with k iterations requires approximately $(k+1) \times 10.000$ flops (1 fit for the initial guess and k remaining fits for each iteration).

3. A semi-parametric fit with 5 "profile points" costs 2.000 flops in low resolution, while in high resolution 14 "profile points" are computed requiring 15.000 flops.

4. A single semi-parametric fit is approximately four times faster than a Gaussian fit in low and high resolution. However, one should keep in mind that a semi-parametric fit in general can not be performed directly without any preceding center search by a M-estimator of location.

5. Let $n \times n$ be the dimension of the input patch, i.e. $n = 7/n = 16$ for low/high resolution. While the computing time for the Gaussian fit will increase with $O(n^2)$, the computing time for a semi-parametric fit will increase with $O(n^2 \cdot \log(n))$. The reason is that a Gaussian fit basically sums over all data points while sorting algorithms are needed for a semi-parametric fit.

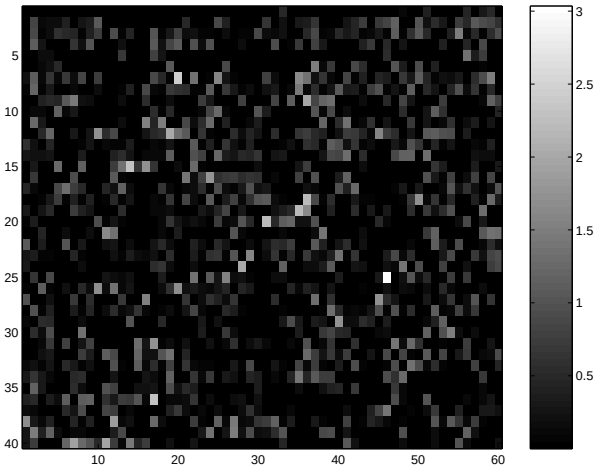


Figure 14: Relative error of the 40×60 spots

6. On a Pentium II with 400 MHz 1.000.000 flops take approximate 7.3 sec.

7. On a Pentium II one field with 80×120 spot locations, moderately filled takes half to one hour in Matlab.

8. On a Pentium II one entire ONF image with 240×240 spot locations, moderately filled takes three to six hours in Matlab.

9. An already implemented C-version (Khoros API) of the non-robust spot fitting with subtracting neighbors for all spots needs only about four minutes on the same machine (including the grid fitting).

6 Conclusion

The basic problems in spot fitting are overlapping and non Gaussian spots. Overlaps with up to three or four neighbors can be reliably solved by robust fitting and subtracting neighboring models with a subsequent refit. For overlapping situations with more than four neighbors too few consistent data is available for robust estimators. One should remember that the highest possible breakdown point of a robust estimator is $\epsilon^* = 0.5$. In such a case one should avoid any fitting and assign a "standard spot model" to the spot location and do the first fit after subtracting the neighbors. Such a "standard spot model" can be computed by first estimating the center by M-estimation of location, second taking a 'standard' dispersion matrix (the spots are approximately of equal size) and estimating the amplitude by least square.

7 Outlook

We provide the following suggestions for the future work:

Finding Better Models The parametric fit depends on its model. We observed in our experiments – especially in high-resolution images - spots which have no Gaussian distribution. One may adapt the Gaussian or construct an alternative model. On the other hand, it is possible to save time by reducing the parameter space, i.e. taking only rotation symmetric models.

Finding alternative measures for goodness of fit and detection limit We are rather satisfied with the introduced measures for goodness of fit and detection limit. However, one may find an alternative approach by constructing other statistics or using a (maximum) entropy method for detection.

Contructing confidence intervals for parameters In statistics it is common to give confidence intervals, ellipses, etc. for the parameters or even for the model. For our problem it would be useful not only to have confidence intervals for the parameters but also for the volume. Assuming a given center or a perfectly determined center a confidence interval for the volume can be directly constructed from a confidence interval for the amplitude and dispersion matrix.

Developing machine learning algorithms When analyzing a ONF-library the computer computes over 4600 Million fits. However, the computer does not learn what is a good fit or what is a spot. It does not learn that a certain volume estimation cannot be possible. The computer should adapt to new conditions and be more fault-tolerant. Robust estimation, detection limit, fit acceptance depend on parameters the prior choice of which may not stay optimal from image to image, from library to library or even from experiment to experiment. Furthermore, the computer could develop some heuristics like: the Gaussian fit always overestimates the volume by 10%.

Other applications The results of this work can be adapted to solve problems in other fields of science. Direct application with few modifications can be done for detection and quantification of galaxies or Braille recognition. Furthermore, the techniques introduced can used for object recognition or reconstruction and robust vision.

References

[1] N. Brändle, H. Lapp, and H. Bischof. Automatic Grid Fitting for Genetic Spot Array Images Containing Guide Spots. In *8th Intl. Conf. on Computer Analysis of Images and Patterns, Ljubljana, Slovenia, September 1–3*, pages 357–366, 1999.

[2] M. Chee *et al*. Accessing Genetic Information with High-Density DNA Arrays. *Science*, 274:610–614, 1996.

[3] J. Hartung: Multivariate Statistik, R. Oldenburg Verlag München Wien, 1989.

[4] R. J. Johnston *et al*. Autoradiography using storage phosphor technology. *Electrophoresis*, 11:355–360, 1990.

[5] Benjamin Lewin. *Genes VI*. Oxford University Press, 1997.

[6] R. A. Maronna. Robust M-estimators of multivariate location and scatter. *The Annals of Statistics*, 4(1):51–67, 1976.

[7] S. Meier-Ewert *et al*. An automated approach to generating expressed sequence catalogues. *Nature*, 361:375–376, 1993.

[8] H. J. Noordmans and A. W. M. Smeulders. Detection and Characterization of Isolated and Overlapping Spots. *Computer Vision and Image Understanding*, 70(1):23–35, 1998.