Universität des Saarlandes

Fachrichtung 6.1 – Mathematik

Preprint Nr. 301

Multi-Class Electrostatic Halftoning

Christian Schmaltz, Pascal Gwosdek and Joachim Weickert

Saarbrücken 2011

Fachrichtung 6.1 – Mathematik
Universität des Saarlandes

Preprint No. 301
submitted: October 28, 2011

Multi-Class Electrostatic Halftoning

Christian Schmaltz
Mathematical Image Analysis Group
Faculty of Mathematics and Computer Science
Saarland University, Campus E1.1
66041 Saarbrücken
Germany
schmaltz@mia.uni-saarland.de

Pascal Gwosdek
Mathematical Image Analysis Group
Faculty of Mathematics and Computer Science
Saarland University, Campus E1.1
66041 Saarbrücken
Germany
gwosdek@mia.uni-saarland.de

Joachim Weickert
Mathematical Image Analysis Group
Faculty of Mathematics and Computer Science
Saarland University, Campus E1.1
66041 Saarbrücken
Germany
weickert@mia.uni-saarland.de

Edited by
FR 6.1 – Mathematik
Universität des Saarlandes
Postfach 15 11 50
66041 Saarbrücken
Germany

Fax: + 49 681 302 4443
e-Mail: preprint@math.uni-sb.de
WWW: http://www.math.uni-sb.de/

Multi-Class Electrostatic Halftoning

Christian Schmaltz Pascal Gwosdek Joachim Weickert

October 28, 2011

Abstract

Electrostatic halftoning, a sampling algorithm based on electrostatic principles, is among the leading methods for stippling, dithering, and sampling. However, this approach is only applicable for a single class of dots with a uniform size and colour. In our work, we complement these ideas by advanced features for real-world applications. We propose a versatile framework for colour halftoning, hatching, and multi-class importance sampling with individual weights. An additional model-based ink coverage analysis improves the visual result in very dark image regions. Our novel approach is the first method that globally optimises the distribution of different objects in varying sizes relative to multiple given density functions. The quality, versatility, and adaptability of our approach is demonstrated in various experiments.

1 Introduction

Many applications in computer graphics and image processing require image-adaptive *importance sampling* strategies, which arrange a finite number of points according to a given continuous density function. In the context of printing or non-photorealistic rendering, this process is called *halftoning* and describes the placement of inkblots onto paper [Kip01, Sec02]. It is also possible to use more complex primitives, in which case the same idea is known as *object placement* [DHL*98, Wei10]. Furthermore, importance sampling is widely employed as a component in a large number of more advanced applications. Examples include *numerical integration* such as Quasi-Monte Carlo methods [Hal60, Coo86], *geometry processing* [SAG03], *ray-tracing* [PH04], and *image-based lighting* using high dynamic range images [KK03].

In the literature, a lot of different algorithms can be found to solve the fundamental problem described above. Early works include the introduction of non-uniform sampling for ray-tracing [DW85], and dart throwing techniques [Coo86].

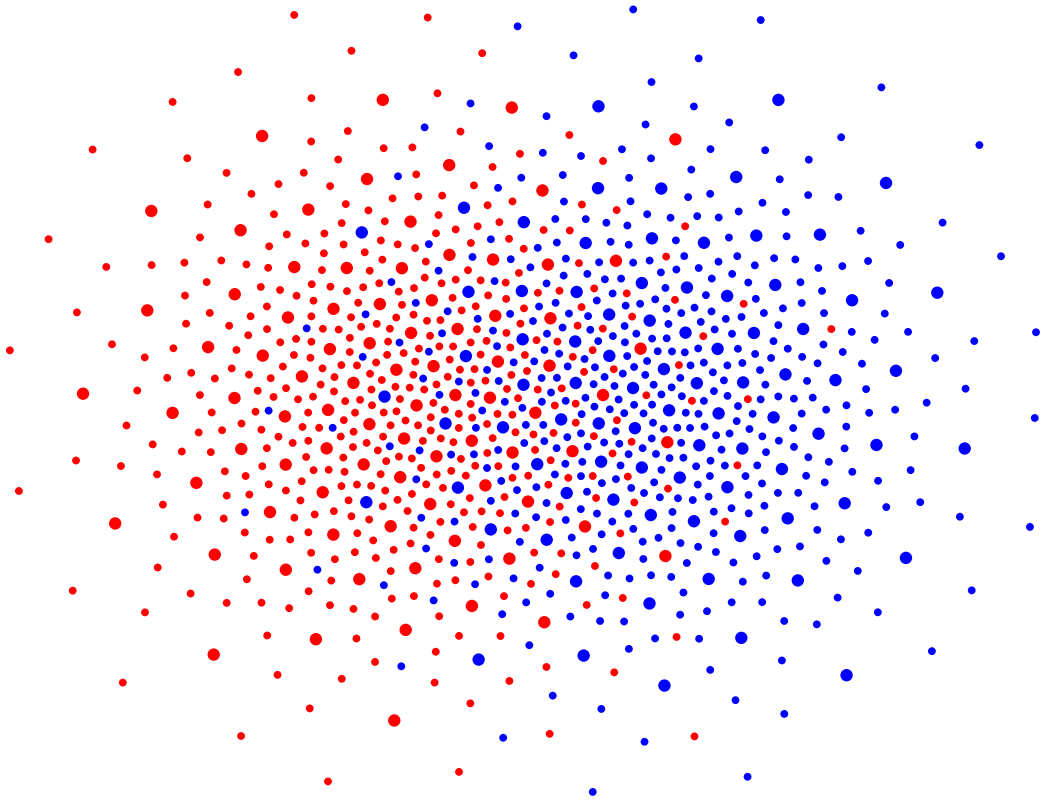

Figure 1: Example halftoning result with multiple classes.

The latter algorithm gave rise to a large number of new algorithms and acceleration schemes such as *relaxation dart throwing* [MF92] and even to a parallel dart throwing algorithm for sampling complex 3-D manifolds [BWWM10].
Due to their speed, algorithms based on tilings are also very popular. In [ODJ04], a recursive Penrose tiling is used, while a polyomino tiling is employed in [VO08]. Several other fast approaches employ Wang tiles [Wan61, HDK01], or even progressive recursive Wang tiles [KCDL06]. Another possibility is to utilise corner tiles [LD06] with a fixed set of tiles. We refer to the survey by Lagae and Dutré [LD08] for more details.
A widely used but much slower approach is Lloyd's method, which was advocated in computer graphics by [MF92]. Lloyd's method must be manually stopped before convergence if regularity artefacts are undesired, though. Recently, this drawback was eliminated by capacity constraints Voronoi tesselations [BSD09]. This also allows to approximate the image – or more generally any underlying probability density function (*PDF*) – much more accurately. A faster version of capacity constrained Voronoi tesselations was introduced by Lie *et al.* [LNW*09a, LNW*09b].

In [SGBW10], sampling points are modelled as charged particles moving in an electric force field induced by the underlying PDF. The resulting steady-state was shown to yield an even higher approximation quality than the approach from [BSD09]. Due to the fact that this so-called *electrostatic halftoning* is easy to implement on parallel hardware such as graphics cards, results are also available faster. This is especially true when using the fast summation algorithm on GPUs from [GSWT11]. [Fat11] applies ideas closely related to those in [SGBW10], but employs truncated Gaussian kernels instead of Coulomb potentials. By using localised kernels and a multi-scale minimisation approach, even a linear runtime is achieved. This seems to come at the expense of a slightly lower approximation quality and a missing adjustability of the exact number of points.
Moreover, there are numerous areas which are not concerned with selecting appropriate point positions, but apply similar ideas in the context of more complex applications. One example is hatching, in which the task is to represent an image by lines (instead of inkblots). The results of such approaches closely resemble engravings and manual drawings commonly found in old medical textbooks. Nowadays, this is often achieved by rendering 3-D models [WS94, PHWF01, ZISS04]. However, there are also works that can use a single 2-D image as input: In [PB94], an approach for creating digital engravings via an Eikonal equation is presented. In [JEGPO02], a sample training patch of strokes is employed to guide the algorithm. Secord *et al.* [SHS02] choose the x and y coordinates of primitives after each other by employing cumulative density functions of the image. However, these approaches only use a single kind of primitives.
Another example is the challenging case of multi-class sampling. Here, points are annotated by class and size attributes, according to which they must be distributed. This has numerous applications, such as the placement of heterogeneous objects, printing in different colours, or second-order screening. In such cases, the behaviour is steered by different, potentially conflicting PDFs between which a compromise must be found. One prominent approach to handle different types was proposed by Wei [Wei10]. Since it adapts dart throwing to the new setting, it is very fast but suffers from a mediocre approximation quality. However, to the best of our knowledge, the general problem of placing differently coloured, potentially non-circular primitives in different sizes was not discussed in the literature so far.
In this work, we extend the electrostatic halftoning approach, which was introduced by [SGBW10], to cope with the problems described above. Our contributions are the following: Firstly, we illustrate how to improve visual results in dark image areas. This is done by adjusting the underlying image before halftoning to account for overlapping inkblots. Secondly, we introduce the changes necessary to deal with inkblots of different sizes. This also allows to create second order screening results similar to those used in the printing industry. Thirdly, we give

details on how to create halftones for output devices with arbitrary colour models, including asymmetric models such as the CMY-RB-K colour model. As a fourth contribution, we present our approach to multi-class sampling using different objects with varying sizes. An example for such a setup is shown in Figure 1. Our fifth and last contribution is a novel approach for hatching, which can even be combined with any of the other contributions.
Our paper is organised as follows: We start with a short repetition of the electrostatic halftoning algorithm in Section 2. Section 3 introduces our improved approach for handling very saturated image regions. We further extent the basic approach to second order screening techniques using inkblots with different sizes in Section 4. In Section 5, we give details about how to halftone images in arbitrary colour models, and illustrate how to perform multi-class sampling. Section 6 explicates how to (additionally) use lines instead of inkblots to approximate images. After performing a detailed experimental evaluation in Section 7, we conclude the paper in Section 8.

2 Electrostatic Halftoning

The idea behind electrostatic halftoning is to model black dots (or sampling points) by infinitesimally small negatively charged particles moving in a pure 2-D world [SGBW10]. Let N be the total number of particles. One identifies each particle $m \in \mathcal{P} := \{1, \ldots, N\}$ only by its position $\boldsymbol{p}_m$ and its charge q_m. Due to electrostatic forces, the particles repel each other. The force acting on the particle m due to the other particles is given by

$$\boldsymbol{F}_m^{(R)} = -\sum_{\substack{n \in P \\ n \neq m}} \frac{k q_m q_n}{\|\boldsymbol{p}_n - \boldsymbol{p}_m\|^2} (\boldsymbol{p}_n - \boldsymbol{p}_m) \,, \tag{1}$$

where k is a constant whose value is irrelevant.
Additionally, the particles are attracted by the underlying image (or density function) u, whereby image regions attract particles proportional to the value of u. For halftoning a grey-valued image, for example, $u(x,y)$ might equal $1.0 - f(x,y)$, where $f(x,y)$ is the grey value of the image taken from the interval $[0,1]$.
Each image point $\boldsymbol{x} := (x,y)$ is assumed to be positively charged with a charge that equals $u(x,y)$. This results in an attractive force acting on the m-th particle proportional to

$$\boldsymbol{F}_m^{(A)} = \int_\Omega \frac{k q_m u(\boldsymbol{x})}{\|\boldsymbol{x} - \boldsymbol{p}_m\|^2} (\boldsymbol{x} - \boldsymbol{p}_m) \, \mathrm{d}\boldsymbol{x} \,. \tag{2}$$

If u is only given at discrete positions, as it is the case if u is an image, the integral is replaced by a discrete sum.
In this equation, we assume that the integral (or sum) over all values in u equals N such that the total amount of positive and negative charges is identical. This automatically binds particles to the image domain. If this assumption is violated, one only has to multiply u with an appropriate constant.
Note that no velocity or even acceleration of the particles is modelled. Instead, the total displacement of each particle is directly obtained by adding the forces $\boldsymbol{F}_m^{(R)}$ and $\boldsymbol{F}_m^{(A)}$. The resulting displacement is scaled by an "artificial time step size" τ, resulting in:

$$\boldsymbol{p}_m^{(new)} = \boldsymbol{p}_m + \tau \left(\boldsymbol{F}_m^{(R)} + \boldsymbol{F}_m^{(A)} \right) \tag{3}$$

This update equation can be regarded as performing a gradient descent of the potential energy in the underlying particle system. As in [SGBW10], we set $\tau k = 0.1$ and use a direct summation algorithm implemented using NVidia's CUDA [NVi11]. However, a faster algorithm employing an NFFT (non-equispaced fast Fourier transform) was also recently proposed [TSG*11] and ported to GPUs [GSWT11].
To prevent the algorithm from getting stuck in local minima, a simulated annealing strategy called "shaking" was proposed in [SGBW10]. This strategy is also included in our implementation. However, since random number generators are available on a per-thread basis nowadays, we can perform this step directly on the GPU.
Note that the attractive image force only depends on the position $\boldsymbol{x}$ and can therefore be precomputed for the regular grid underlying the image. The attractive image force at positions between grid locations are approximated by bilinear interpolation. This automatically smoothes the attraction field, and thus prevents possible problems introduced by the discrete nature of the underlying image.

3 Grey-Value Correction

As a first enhancement of the aforementioned model, we propose a novel approach to accurately represent dark image areas. In the classical model, those regions are often rendered too bright, which is a result of the infeasibility to tessellate regions with circular discs of constant size. In such cases, discs are overlapping and leave the corresponding area as bright spots.
The classic halftoning literature approaches this problem by using circumcircles of regular hexagons [Uli87] instead of circles that cover the same area as the

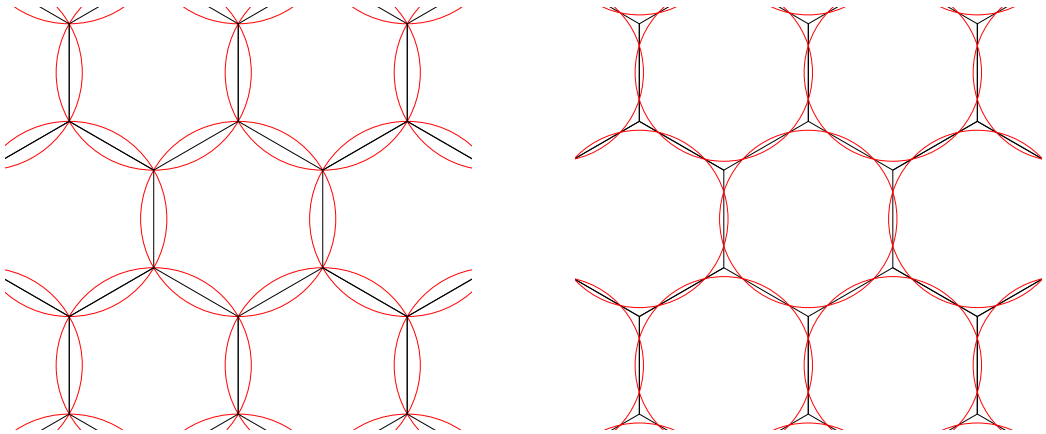

Figure 2: Dot sizes (red circles) with respect to the underlying hexagon structure. **Left**: Classical approach using circumcircles. **Right**: Our approach using circles with the same size as the underlying hexagons.

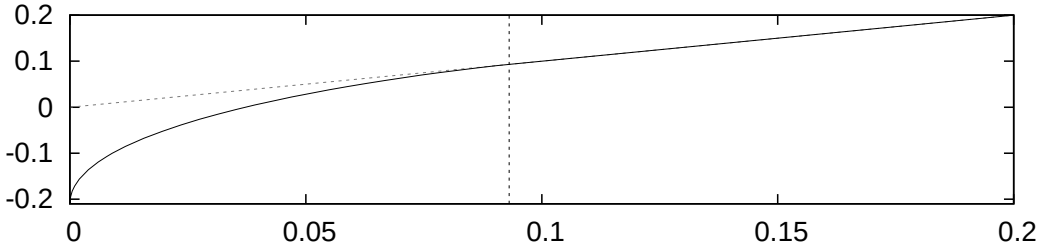

Figure 3: Plot of the tonemapping operator $T(x)$. The dotted lines indicate the identity function and the value $1-\frac{\pi}{2\sqrt{3}}$ after which $T(x)=x$ holds.

hexagon, see Figure 2. Thus, each partition of space is covered by at least one circle in a regular honeycomb pattern. Since this measure darkens the whole image in brighter regions, a tonemapping operator is applied to the input prior to halftoning. It artificially brightens up areas such that the 'wrong' halftone using larger circles again well approximates the original.

In this paper, we propose a similar idea to prevent the problem described above. However, instead of brightening all image regions, we darken only very dark image areas. Consequently, more particles are attracted to regions that previously appeared too bright. This results in a stronger overlap, and thus a better approximation.

Let us assume that particles are arranged in a regular, i.e. energetically optimal, honeycomb pattern. In order to make this arrangement resemble the desired grey value, we propose the tonemapping operator shown in Figure 3:

$$T(x) = \begin{cases} 1 - \frac{\pi}{2\sqrt{3}\,c^2}, & x \leqslant 1 - \frac{\pi}{2\sqrt{3}} \\ x, & \text{else} \end{cases} \tag{4}$$

with c chosen such that

$$\frac{\pi}{2\sqrt{3}\,c^2}\left(1 - \frac{6}{\pi}\left(\arctan\left(\sqrt{\frac{1}{c^2} - 1}\right) - c\,\sqrt{1-c^2}\right)\right) = x\,, \tag{5}$$

$$\frac{\sqrt{3}}{2} \leqslant c \leqslant 1\,. \tag{6}$$

If this operator is used, it can be shown by computing the overlap between adjacent inkblots that the halftoning of $T(u(x))$ has the same average grey-value as the original image $u(x)$.

As can be easily seen, tone mapping is necessary if and only if the grey value is smaller than $1 - \frac{\pi}{2\sqrt{3}} \approx 0.0931$, whereby the image is assumed to have a range of $[0,1]$. For an image with 256 possible grey levels, tone mapping must therefore be applied for only 24 grey levels, since $0.0931 \cdot 255 < 24$ holds, while all but one grey values must be adapted when using circumcircles.

The only challenging part left is the evaluation of (5), which has to be solved for c depending on x. This can easily be done once by the use of a computer algebra system, as images only contain few values due to quantisation. A table for a grey range of 8 bit depth, as well as the complete derivation of (5), is available at our supplementary material website: `http://www.mia.uni-saarland.de/Research/Electrostatic_Halftoning/multi/index.shtml`

4 Second Order Screening

Based on this improved model for electrostatic halftoning, we now propose a versatile framework for multi-class sampling. Our approach allows to handle points of different colour, size, and shape. As the first step towards this goal, we illustrate the generalisation to points in different sizes.

Our approach bases on a simple idea: When assigning a charge proportional to the size of the inkblot to the corresponding particle, inkblots are automatically distributed according to their size. Although the underlying image is approximated well with this simple approach, larger inkblots may accumulate, see the left image in Figure 4. This gives a perturbing impression of the image, and is also bad for printing applications, in which some (or even all) of the smaller inkblots may not be printed due to technical reasons.

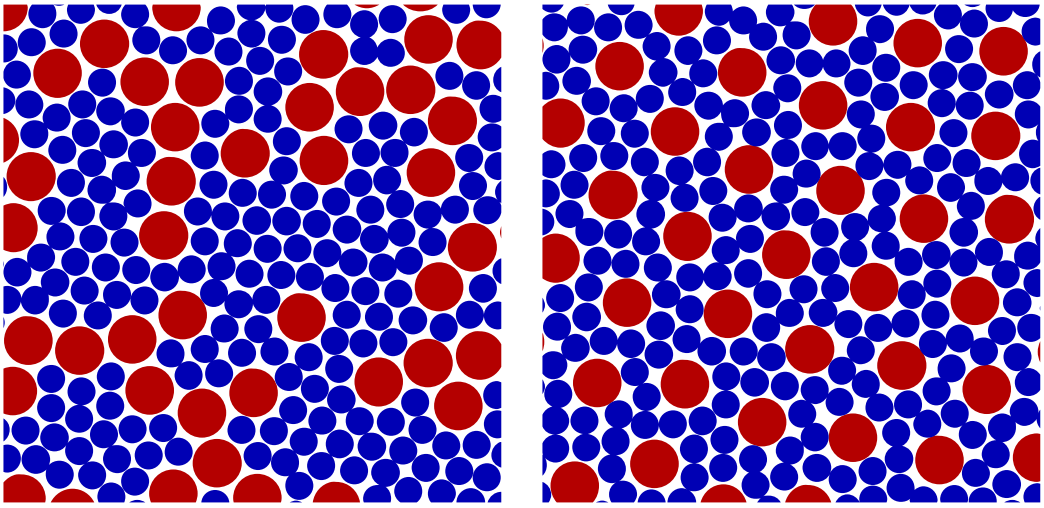

Figure 4: Comparison of different repulsion models for second order screening of a constant image. Shown are results without (left, $C = 1$) and with (right, $C = 2$) the additional force responsible for an equal distribution of large particles. The two colours are for illustration purposes only.

To ease the explanation how to prevent this problem, we first consider a simple case with two different particle sizes. In this situation, we require the larger particles to additionally approximate the complete image well, as shown in the right image in Figure 4. Then, the halftoning result is a good approximation of the image both with and without the smaller inkblots.
This can be achieved by introducing a supplementary force that ensures a good distribution of the large particles: In addition to the repulsive forces from all particles, we introduce an additional repulsive force acting only between large particles. This additional force is exactly the same force that occurs in our particle system when only large particles are present, except for a multiplicative constant

	S	**L**
S	$A_\mathrm{S}A_\mathrm{S}$	$A_\mathrm{S}A_\mathrm{L}$
L	$A_\mathrm{S}A_\mathrm{L}$	$CA_\mathrm{L}A_\mathrm{L}$
u	A_S	$(w_S + Cw_L)A_\mathrm{L}$

Figure 5: Interaction matrix for the total repulsion strengths between small particles (S) with area A_S and large particles (L) with area A_L. The last line indicates the interaction strengths of the particles with the attractive image field. The constant C is a user-defined variable.

	S	M	L
S	$A_{\mathrm{S}}A_{\mathrm{S}}$	$A_{\mathrm{S}}A_{\mathrm{M}}$	$A_{\mathrm{S}}A_{\mathrm{L}}$
M	$A_{\mathrm{S}}A_{\mathrm{M}}$	$C_1A_{\mathrm{M}}A_{\mathrm{M}}$	$C_1A_{\mathrm{M}}A_{\mathrm{L}}$
L	$A_{\mathrm{S}}A_{\mathrm{L}}$	$C_1A_{\mathrm{M}}A_{\mathrm{L}}$	$C_2\,A_{\mathrm{L}}A_{\mathrm{L}}$
u	A_{S}	$(w_{\mathrm{S}}+C_1(w_{\mathrm{M}}+w_{\mathrm{L}}))A_{\mathrm{M}}$	$(w_{\mathrm{S}}+C_1w_{\mathrm{M}}+C_2w_{\mathrm{L}})A_{\mathrm{L}}$

Figure 6: Interaction matrix i for the total repulsion strengths between small particles (S) with area A_{S}, medium sized particles (M) with area A_{M}, and large particles (L) with area A_{L}. The last line indicates the interaction strengths of the particles with the attractive image field. The variables w_S, w_M, and w_L correspond to the fraction of charges located in small, medium, and large particles, respectively.

$C-1$. As a consequence, large particles repel each other stronger than explained in (1). The constant $C \geq 1$ is used as a tradeoff between the two requirements imposed on the system: A small C leads to a good approximation of the original image with respect to all particles, but to a worse approximation when only large particles are considered. For large C, the converse holds.
To maintain the electric neutrality of the overall image, the electric field induced by the image must also attract large particles stronger. Let us denote the fraction of the negative charges located in large particles by w_L, and the fraction of the negative charges located in small particles by w_S. Then, large particles are attracted C times as strong from the image part corresponding to the large particles, and "normally" from the remainder. Thus, the force attracting large particles is multiplied by the factor $w_S + Cw_L$.
Combining all these concepts results in the *interaction matrix i* depicted in Figure 5 in which each entry $i(A_j, A_k)$ indicates how strong the interaction between particles from the classes A_j and A_k are.
It should be noted that using this additional force slightly worsens the approximation quality of all particles. This is a principal problem which arises from the fact that a tradeoff between two conflicting requirements is searched for. However, as is shown in the experiments section, the impact of this unavoidable effect is often negligible.
It is straightforward to extend these ideas to three (or more) classes. Again, one demands that all the particles approximate the image well. Additionally, this property should also hold when removing the particles with the smallest size, or the smallest sizes, respectively. For three classes, this results in the interaction matrix depicted in Figure 6. In our experiments, we typically choose the resulting parameters C_i in a multiplicative manner, i.e. $C_i = C_1^i$. However, other choices are also admissible depending on the desired result. In general, the attractive image force

strength $f(s)$ on the sth particle is given by

$$f(s) = \frac{\sum_{t \in \mathcal{P}} i(A_s, A_t) A_t}{\sum_{t \in \mathcal{P}} A_t}. \tag{7}$$

As illustrated in Figure 12, our approach can even handle situations in which there are as many classes as particles, i.e. situations in which each particle has a unique size, when choosing the interaction matrix appropriately.

5 Colour Halftoning and Multi-Class Sampling

In [SGBW10], it was suggested to halftone colour images by treating all channels separately. However, this is only possible with linear colour spaces such as CMY and RGB.

Printers typically use more complex colour models, though. A common example is the (comparably simple) CMYK colour model which is found in all sorts of printers ranging from inexpensive inkjet printers such as the Canon Pixma i560 via business-scale colour laser printers like the Kyocera FS-C5200DN up to mass-production printing machines. Some recent printers even use more complex, 'asymmetric' colour models: The Canon Pixma iP8500, for example, uses the CMY-RG-K colour space, while the CMY-RB-K colour space is used in the Stylus Photo R1800 by Epson. In large scale production printing machines, the 'full' CMY-RGB-K colour model is typically used.

In this section, we give details how to extent electrostatic halftoning such that it can handle any of these colour models. To simplify the explanations, we assume that all particles have the same size.

The only challenge when using such colour models is that the corresponding channels are no longer independent. This again allows to use the concept of interaction matrices introduced in the last section. This time, however, each entry in the matrix denotes the interaction between particles with two specific colours instead of with two specific sizes.

The interaction matrix in case of independent channels is simply the identity matrix. As a first example in which non-zero off diagonals are used to handle interacting channels, let us consider the CMYK colour model. As a canonical requirement, each class of particles should approximate the corresponding channel, which is again modelled by an identity matrix. Additionally, we demand that particles from certain classes do not overlap. For the CMYK colour model, neither two particle of the same colour nor a black and a chromatic particle should overlap. The latter requirement arises because a black particle represents a union of all chromatic colours. This is coded in a *non-overlap* matrix. The final interaction matrix is then given by a linear combination of these two matrices:

	C	M	Y	R	G	B	K
C	1	α	α	α	α	α	α
M	α	1	α	α	α	α	α
Y	α	α	1	α	α	α	α
R	α	α	α	1	α	α	α
G	α	α	α	α	1	α	α
B	α	α	α	α	α	1	α
K	α	α	α	α	α	α	1

	C	M	Y	R	G	K
C	1	0	α	α	α	α
M	0	1	α	α	α	α
Y	α	α	1	α	α	α
R	α	α	α	1	α	α
G	α	α	α	α	1	α
K	α	α	α	α	α	1

Figure 7: Interaction matrices for the CMY-RGB-K (left) and the CMY-RG-K (right) colour space.

$$(1-\alpha)\underbrace{\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{pmatrix}}_{\text{identity}}+\alpha\underbrace{\begin{pmatrix}1&0&0&1\\0&1&0&1\\0&0&1&1\\1&1&1&1\end{pmatrix}}_{\text{non-overlap map}}=\underbrace{\begin{pmatrix}1&0&0&\alpha\\0&1&0&\alpha\\0&0&1&\alpha\\\alpha&\alpha&\alpha&1\end{pmatrix}}_{\text{interaction matrix}}$$

In our experiments, we use $\alpha = \frac{1}{2}$, which gives the same weight to both constraints.

It is very easy to extend this approach to other colour models. For example, there are several possibilities for the non-overlap matrix used for the CMY-RGB-K colour model. Since all combinations of chromatic colours are already present, the simplest approach is to forbid any overlap. This is coded in the 7×7 non-overlap matrix containing only 1s. For the CMY-RG-K colour model, it is natural to demand that only cyan and magenta particles can overlap, as this is the only combination yielding a colour that is not contained in other particles, yet. The resulting interaction matrices for these examples are shown in Figure 7.

Next, let us explain how many particles of each colour should be used. In fact, there are many possible choices, including a chromatic decomposition resulting in nonempty C, M, and Y channels only, or an (a)chromatic decomposition with under colour removal (UCR), see [Kip01]. In all our experiments, we use a simple achromatic composition of the CMY colour space, which sets the K channel to the minimum of the three given CMY channels, and subtracts this values from these three channels afterwards. Afterwards, we separate the R channel from the M and Y channels. Similarly, G is separated from C and Y, and B is separated from C and M. Note that this step is unique and commutative for the CMY-RGB-K

	$\mathbf{S}_a$	$\mathbf{L}_a$	$\mathbf{S}_b$	$\mathbf{L}_b$
$\mathbf{S}_a$	$A_\mathrm{S}A_\mathrm{S}$	$A_\mathrm{S}A_\mathrm{L}$	$\alpha A_\mathrm{S}A_\mathrm{S}$	$\alpha\ A_\mathrm{S}A_\mathrm{L}$
$\mathbf{L}_a$	$A_\mathrm{S}A_\mathrm{L}$	$CA_\mathrm{L}A_\mathrm{L}$	$\alpha A_\mathrm{S}A_\mathrm{L}$	$\alpha CA_\mathrm{L}A_\mathrm{L}$
$\mathbf{S}_b$	$\alpha A_\mathrm{S}A_\mathrm{S}$	$\alpha\ A_\mathrm{S}A_\mathrm{L}$	$A_\mathrm{S}A_\mathrm{S}$	$A_\mathrm{S}A_\mathrm{L}$
$\mathbf{L}_b$	$\alpha A_\mathrm{S}A_\mathrm{L}$	$\alpha CA_\mathrm{L}A_\mathrm{L}$	$A_\mathrm{S}A_\mathrm{L}$	$CA_\mathrm{L}A_\mathrm{L}$
$\mathbf{u}_a$	A_S	$(w_{S,a}+Cw_{L,a})A_\mathrm{L}$	αA_S	$\alpha(w_{S,b}+Cw_{L,b})A_\mathrm{L}$
$\mathbf{u}_b$	αA_S	$\alpha(w_{S,a}+Cw_{L,a})A_\mathrm{L}$	A_S	$(w_{S,b}+Cw_{L,b})A_\mathrm{L}$

Figure 8: Interaction matrix for multi-class sampling with two classes a and b, each containing two sizes A_L and A_S.

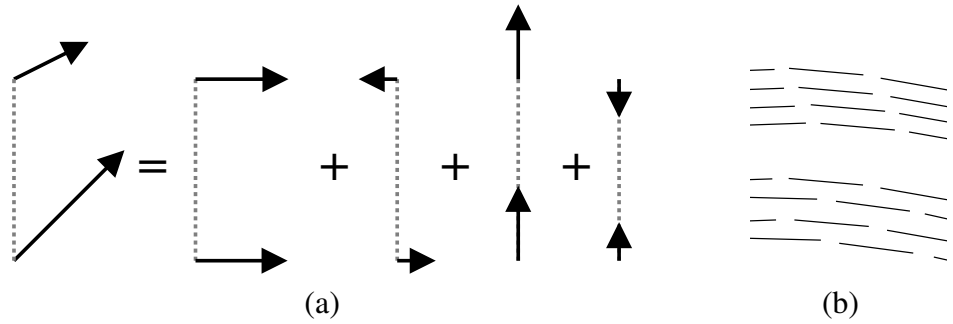

Figure 9: **(a)**: Example on how to split the forces acting on the two ends of a line into different components. The first and third part correspond to translations, the second part to a rotation around the middle of the line, and the last part to a scaling. The strength of each part can easily be found by projecting the forces onto directions perpendicular and parallel to the line and very elementary arithmetic operations. **(b)**: Illustration that an inappropriate line placement can create visible artefacts (top) which are not visible in a good hatching (down).

colour model, and that the introduced interaction matrices also work with any other method to determine the number of particles of each colour.
Finally, we would like to stress that our interaction matrices make it straightforward to combine second order screening and colour halftoning. This does not only allow to create second-order colour images, but also paves the road for multi-class sampling algorithms used for object placement or procedural texture creation. An illustration for the resulting interaction matrix in given in Figure 8, while various examples are shown in Section 7.

6 Hatching

As a last contribution in this paper, we illustrate how to approximate images with lines, i.e. we explain how to generate a hatched version of an image.
The basic idea behind our hatching approach is to approximate each line in the final image by two (or more) connected particles. Starting from an initial configuration, the forces acting on each particle are computed as before, whereas the movement of the lines is determined by the forces acting on each of the particles in the line. To this end, we split these forces into different parts responsible for translations, rotations, and stretching. This is illustrated in Figure 9 for a simple approximation with two particles.
However, the result obtained with this simple approach looks far from pleasing, as the resulting lines have arbitrary lengths and alignments. Thus, we additionally introduce forces that urge lines to achieve the desired (position-depended) length and direction. What line lengths and alignments are actually desired depends on the application. Thus, our GPU hatching implementation simply reads these values from a texture generated on the CPU at the beginning of the program. For the experiments shown here, we set the desired line lengths to a constant, and the desired direction orthogonal to the eigenvector corresponding to the largest eigenvalue of the smoothed structure tensor [FG87]

$$K_\rho * (\nabla u_\sigma \nabla u_\sigma^\top), \qquad \text{where} \qquad u_\sigma := K_\sigma * u \tag{8}$$

where $K_\star$ is a Gaussian with standard deviation $\sigma = 1.0$ or $\rho = 3.0$ times the pixel size, respectively. The width of the lines is chosen such that the average grey value of output and input image are equal.
We compute the weighted average between the two forces responsible to change the direction of each line to obtain the overall rotation. Note that, as in second-order screening and colour halftoning, there are two constraints that should be fulfilled. The appropriate weights can (and should) therefore be chosen by the user depending on whether which is more important: a very accurate approximation of the image, or a perfect match between desired and final line direction. For all our experiments, we set both weights to 0.4. Since the sum of these weights is less than one, this slows down the rotations, i.e. favours moving a line instead of rotating it.
Additionally, it should be noted that there might not be a preferred direction at each position in the image. In this situation, the force responsible to align actual and desired line direction should be additionally weighted by a number in the interval $[0,1]$. In our experiments, we set this number to the largest eigenvalue of the smoothed structure tensor at the current position, divided by the largest eigenvalue occurring anywhere in the image. This way, lines are forced to align

perpendicular to image edges, but may arrange arbitrarily in homogeneous image regions.

Similarly, the forces resulting in a change of the line length are also weighted and averaged. For all experiments presented here, we choose the weights as $\frac{1}{5}$ (force acting on particle) and $\frac{1}{10}$ (force from desired line length), respectively. Due to the fact that the sum of the weights is smaller than 1, the line lengths only change slowly. This is advantageous to prevent particles from overlapping due to these forces, which can result in crossing lines. As this is undesired, one should also prevent lines from crossing when initialising the line positions. We propose to start with very short lines distributed with the same approach as the points in the halftoning algorithm [SGBW10]. Since the line lengths increase only quite slowly, the lines have some time to distribute appropriately before the final line lengths are reached, which allows to prevent basically all crossing lines.

The more particles are used to approximate a line, the better the estimation of the hatching. For the experiments presented in this paper, we use an approximation with three equispaced particles per line, i.e. there is one particle in the middle of each line, and two at their ends, respectively. This already yields very good results at a reasonable running time. Note that forces acting on the middle particle only result in a translation of the line. Thus, this situation is very similar to the two particle case.

Moreover, using (at least) three particles allows to easily prevent the problem illustrated in the upper part of Figure 9(b): If the lines are allowed to distribute arbitrarily, it is possible that the spaces between lines arrange in clearly visible patterns. We simply prevent this problem by assigning a higher charge to the particle in the middle of the line, as explained in Section 4. This automatically encourages the line centres to distribute well. Thus, the situation shown in the lower part of Figure 9(b) is obtained instead of the problematic situation shown in the upper part of that figure. In our experiments, the charge of the middle particle is six times as large as the charge of the particles at the end of each line. However, results are very similar for a large range of charge ratios.

Note that it is also easy to approximate an image with both lines and inkblots, as both are based on the same electrostatic forces. Examples are shown in the experiment section. Similarly, extensions to coloured hatching and combinations of coloured lines and points are also straightforward. In some applications, one might even allow lines to bend. This is possible by replacing the constraint that all points are on one line by a force that only favours this.

Figure 10: Illustration of grey-value correction in dark image areas. Shown are halftoning results of an image containing the word "DARK" with grey value 0 on a background with grey value 10 (320×95). **Top**: Standard halftoning. **Bottom**: With proposed grey-value correction. Note that the difference between both images might be indistinguishable on some output devices, e.g. due to ink bleeding.

7 Experiments

Finally, we illustrate the results obtained with the extensions introduced in each of the previous sections. First of all, we judge the visual effect of the grey-value correction. To this end, we consider an image containing the word "DARK" with grey value 0 on a background with grey value 10, and halftone it with and without this extension. The results and magnifications thereof are shown in Figure 10. When viewing this image on an output device not susceptible to ink bleeding or related problems, one can easily see that, without the proposed extension, the dots do not completely cover the paper in image regions that should be completely black. Consequently, these regions are brighter than they are supposed to be. This is no longer the case using our novel approach, as visible in the bottom of the figure.

In the left column of Figure 11, we show a halftoning of a Gaussian with the two particle sizes 1 and 3 pixels, respectively. In accordance with the simple experiment depicted in Figure 4, both the large particles as well as all inkblots approximate the underlying image well. For this experiment, we have chosen the number of particles of each class in such a way that $w_L = \frac{1}{3}$ and $w_S = \frac{2}{3}$ holds. The right column of the image shows a halftoning of the image "skull" with inkblots of area 1, 2, and 4 pixels with $w_L = \frac{1}{9}$, $w_M = \frac{2}{9}$, and $w_S = \frac{6}{9}$. Again, the image is approximated very well by the set of all particles as well as by the appropriate subsets.

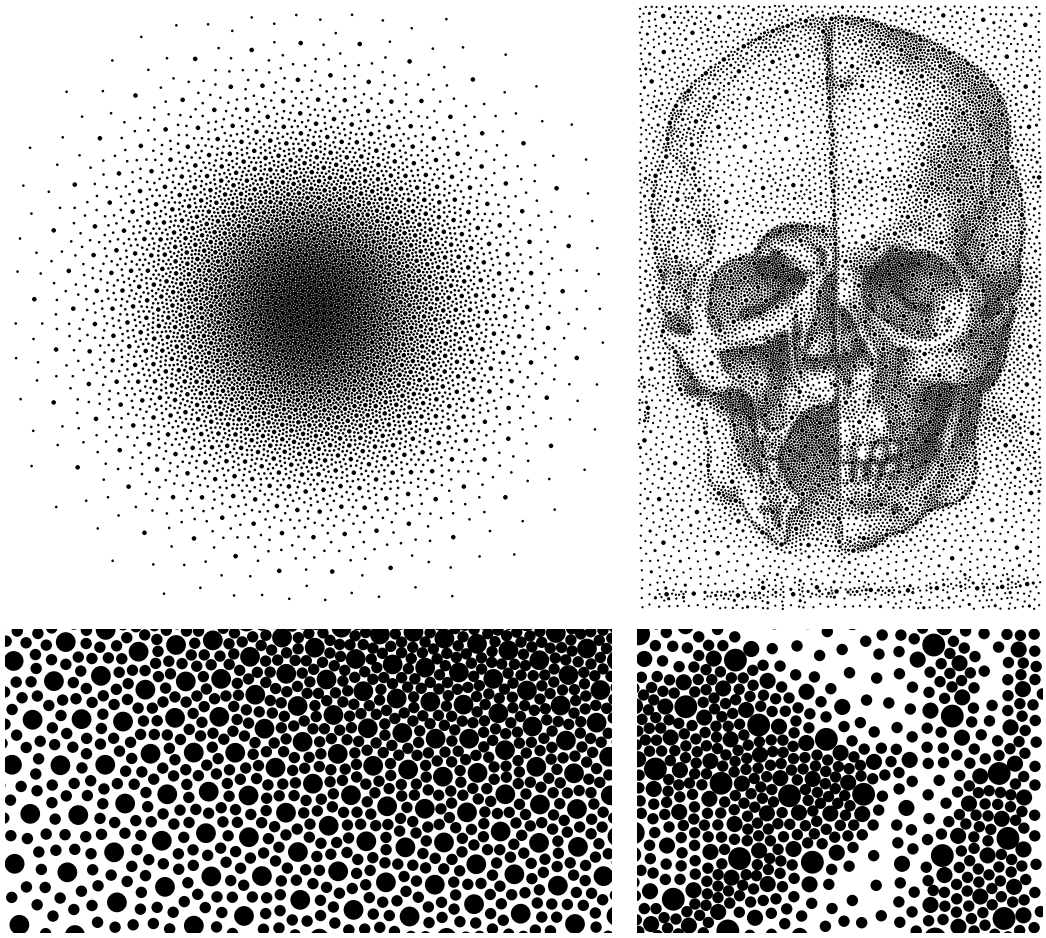

Figure 11: Example halftoning results with second order (left) and third order (right) screening techniques obtained with our algorithm. **Left**: Halftoning of "Gaussian" (256×256). **Right**: Halftoning of "skull" (200×300). **Top**: Complete images. **Bottom**: Zoom into the Gaussian and into the tip of the right eye, respectively.

Figure 12: Halftoning result with dots of continuously varying sizes for the image "trui" (256×256). In all but the first image, some of the dots with smallest radius were omitted. **Top left**: Result with all dots. **Top right**: Result when omitting 25% of the dots. **Bottom left**: Result when omitting 50% of the dots. **Bottom right**: Result when omitting 75% of the dots. Even though the image gets noticeably brighter, it still approximates the underlying image well.

To further illustrate this point, we halftoned the image "trui" using inkblots with unique sizes linearly going from 0.5 to 2 pixels. The resulting halftone, as well as results in which 25%, 50%, and 75% of the particles with smallest size are omitted are shown in Figure 12. For this experiment, we used an interaction matrix with $i(A_j,A_k) = \min(A_j^2,A_k^2)A_jA_k$ for two particles with sizes A_j and A_k. Apart from the inevitable facts that the image becomes brighter and less detailed when omitting particles, one clearly sees that each set of particles still approximates the underlying image well.

Next, we show halftoning results on colour images using different colour models. In Figure 13, results with the linear colour space CMY used in [SGBW10], with the common CMYK colour space, and with the asymmetric CMY-RB-K colour space are shown. Even though none of these colour spaces contains green as a single colour, the dark green T-shirt is represented well in all results.

Figure 14 illustrates the influence of the parameter α on the tradeoff between good representations of the total set of particles and good representations of subclasses in multi-class sampling applications. Small α cause clusters in the subsets corresponding to single classes, while large α generate clusters in the total set to improve the approximation of the single classes. Depending on which representation is more important, α should therefore be chosen application dependent.

As explained before, a similar tradeoff exists in multi-class sampling, which can be steered using the parameter C. Since experiments demonstrating this influence are comparable to Figure 14, we left them out to save space. However, they are available on our supplementary material website.

Figure 15 shows halftoning results for second order multi-class sampling. Due to the large number of constraints between which a compromise must be found, it is very hard to find a good sampling. Nevertheless, our results are quite smooth and do not exhibit striking artefacts which hint at the presence of a second class, or particles with a different size, respectively. Note that, as before, it is still possible to adapt the free parameters C and α if we want to increase the smoothness of specific classes.

Next, we illustrate the visual performance of the proposed hatching algorithm. To this end, we hatched the image "skull" with 20 000 lines with a desired length of 2 and 4 pixels, see Figure 16. In order to improve the visual appearance, edge enhancement via unsharp masking has been performed on the input image before hatching. As a result, even most small details are well visible in the resulting image.

Figure 17 presents images in which both lines and inkblots are used. Shown are results with and without edge enhancement, and with edge enhancement only for the inkblots, but not for the lines. While the latter is likely to be problematic for most other halftoning algorithms, it is straightforward in our approach: We simply precompute the attractive image force with and without edge enhancement, and

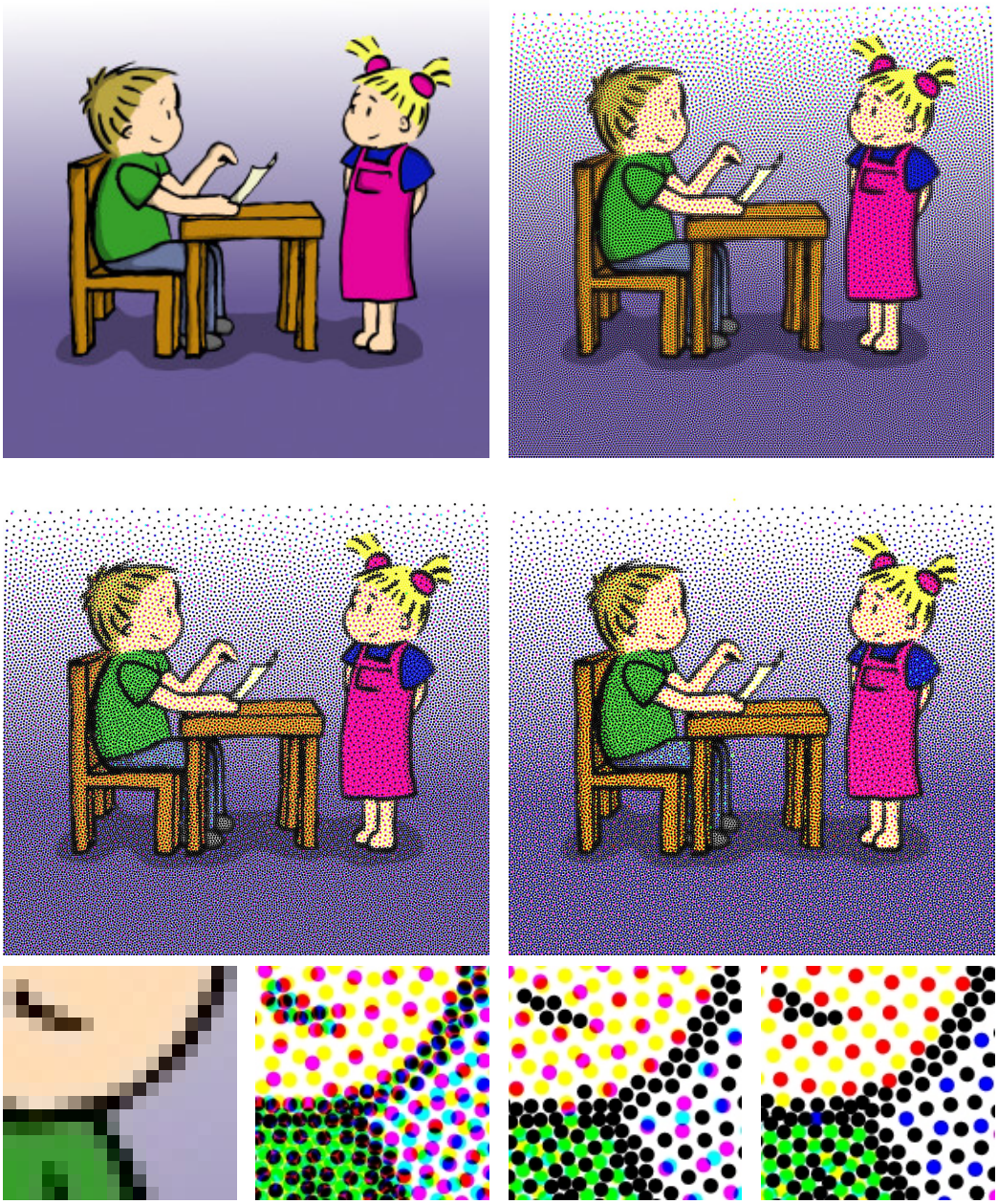

Figure 13: Colour halftoning of the image “comic”. **Top left**: Original image (256×256) **Top right**: Using the CMY colour model. **Middle left**: Using the CMYK colour model. **Middle right**: Using the asymmetric CMY-RB-K colour model. **Bottom row**: Zooms into the images.

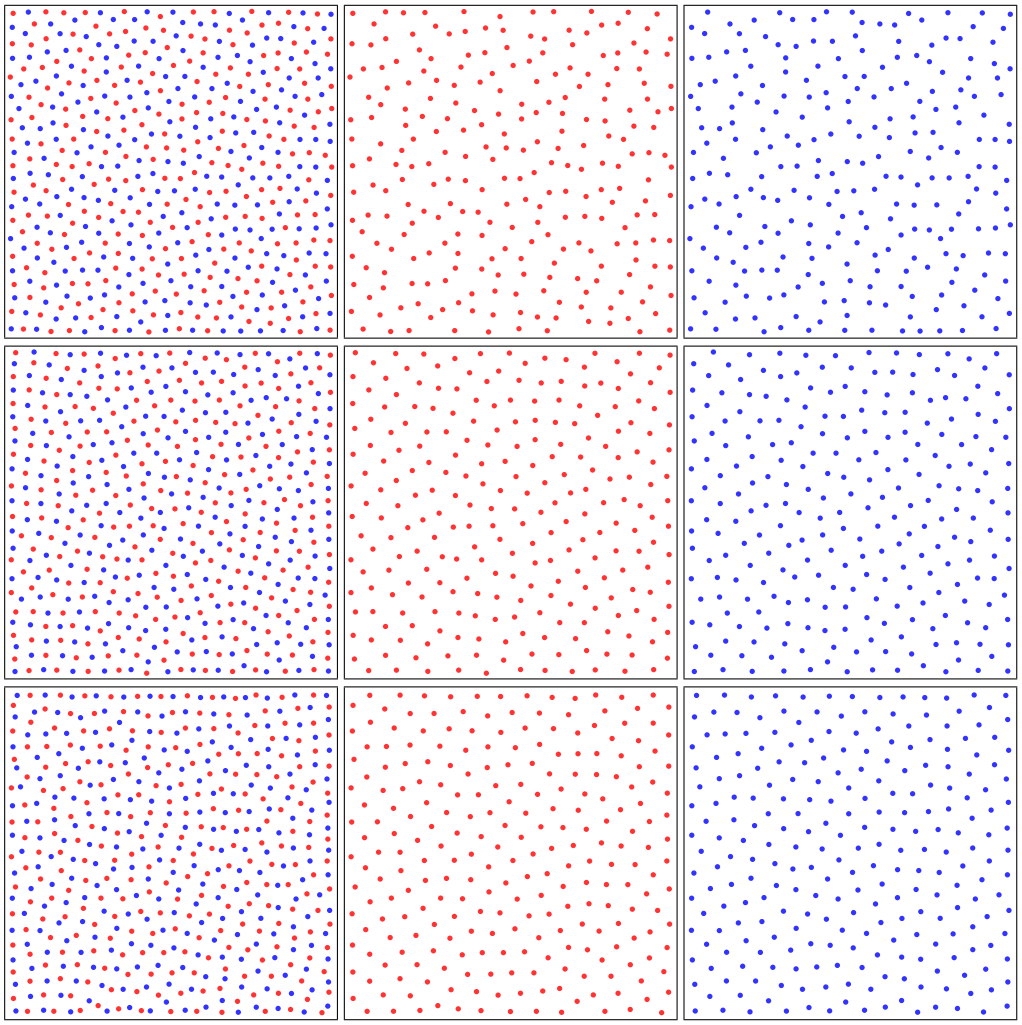

Figure 14: Influence of the weight α on single order multi-class sampling of a uniform image. Shown are, from top to bottom, the total set and the individual sets for $\alpha \in \{\frac{1}{6}, \frac{3}{6}, \frac{5}{6}\}$.

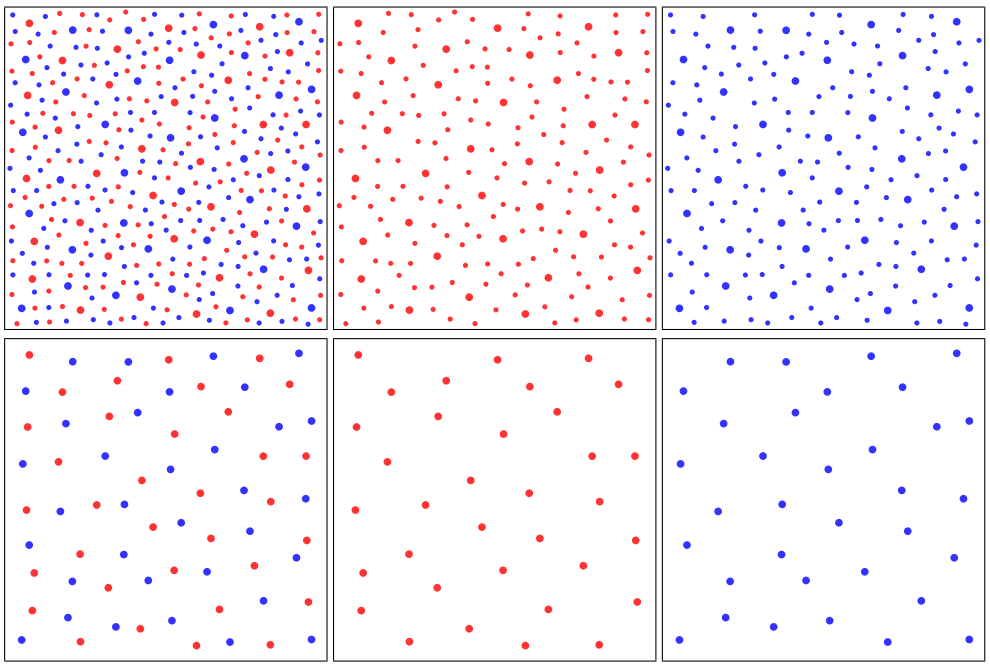

Figure 15: Second order multi-class sampling of a uniform density. **Top left**: Complete result, **Top middle and right**: Red and blue subsets, **Bottom row**: Large particles only.

use the appropriate force in the calculations.
Finally, real-world example images created with our multi-class, multi-size halftoning algorithm are shown in Figures 18 and 19. The first figure shows a forest with two types of trees with two sizes each in which the tree positions are determined with our algorithm. In the second figure, three kinds of ingredients with three sizes each are put on a pizza. By combining hatching with multi-class sampling, it is also possible to use line-like ingredients such as chillies or anchovies.

8 Conclusion

In this paper, we generalised electrostatic halftoning for various real-world applications. We introduced a versatile framework based on interaction matrices, which allows to handle particles in an arbitrary number of colours and sizes. Our approach is the first method to optimise the distribution of heterogeneous points simultaneously based on a few freely adjustable and intuitive parameters. The resulting adaptive sampling method enjoys a broad range of applications, such as object placement, texture generation, or re-lighting. Printing processes benefit from these strategies by a customised rendering process featuring arbitrary colour

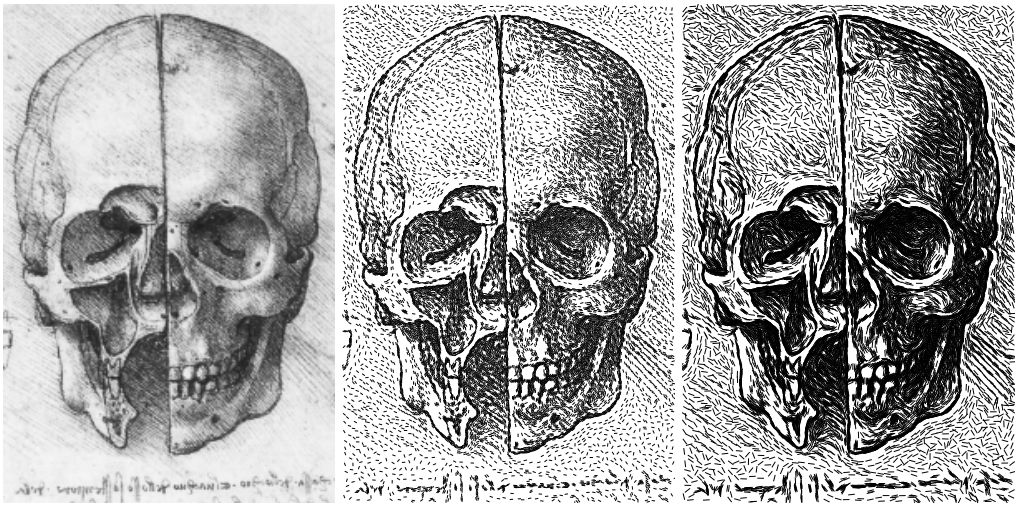

Figure 16: Hatching result of "skull" (200×300). **Left**: Original. **Middle**: Result with a line length of 2 pixels. **Right**: Result with a line length of 4 pixels.

models and dot sizes. This is complemented by a model-based ink coverage correction which enhances the tonal accuracy of high-resolution halftones. The high quality of our flexible framework carries over to anisotropic rendering primitives. Besides sampling with elongated objects, it allows for the generation of hatched images with fine details. All proposed ideas are freely combinable to solve challenging problems with a great simplicity. To this end, our work offers an intuitive methodology to transfer the high quality of electrostatic halftoning to a large range of exciting applications in various areas.

References

[BSD09] BALZER M., SCHLÖMER T., DEUSSEN O.: Capacity-constrained point distributions: A variant of Lloyd's method. *ACM Transactions on Graphics 28*, 3 (2009), 86:1–8.

[BWWM10] BOWERS J., WANG R., WEI L.-Y., MALETZ D.: Parallel poisson disk sampling with spectrum analysis on surfaces. *ACM Transactions on Graphics 29*, 6 (2010), 166:1–166:10. SIGGRAPH Asia.

[Coo86] COOK R. L.: Stochastic sampling in computer graphics. *ACM Transactions on Graphics 5*, 1 (1986), 51–72.

[DHL*98] DEUSSEN O., HANRAHAN P., LINTERMANN B., MĚCH R., PHARR M., PRUSINKIEWICZ P.: Realistic modeling and render-

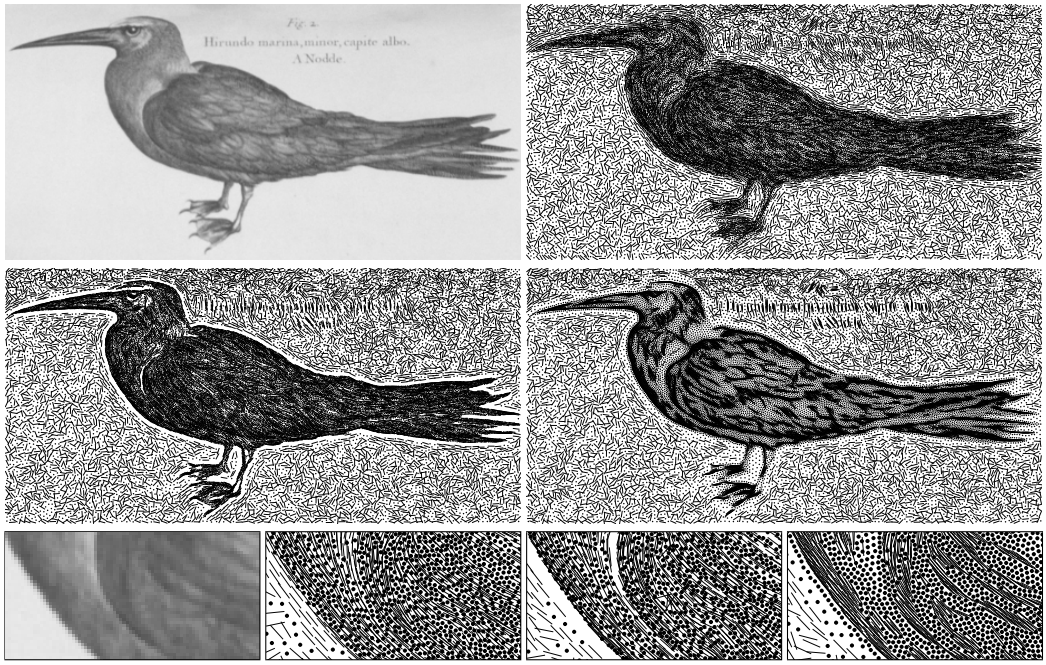

Figure 17: Example of using 10 000 lines and 10 000 dots to approximate the image "bird" (463×229). **Top left**: Original. **Top right**: Result without edge enhancement. **Middle left**: Result with edge enhancement. **Middle right**: Result with edge enhancement only for the lines. **Bottom**: Magnifications to chest region, respectively.

ing of plant ecosystems. In *Proc. SIGGRAPH '98* (New York, NY, USA, 1998), ACM, pp. 275–286.

[DW85] DIPPÉ M. A. Z., WOLD E. H.: Antialiasing through stochastic sampling. In *Proc. 12th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1985), SIGGRAPH '85, ACM, pp. 69–78.

[Fat11] FATTAL R.: Blue-noise point sampling using kernel density model. In *Proc. SIGGRAPH 2011* (New York, NY, USA, 2011), vol. 28, ACM.

[FG87] FÖRSTNER W., GÜLCH E.: A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *Proc. ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data* (Interlaken, Switzerland, June 1987), pp. 281–305.

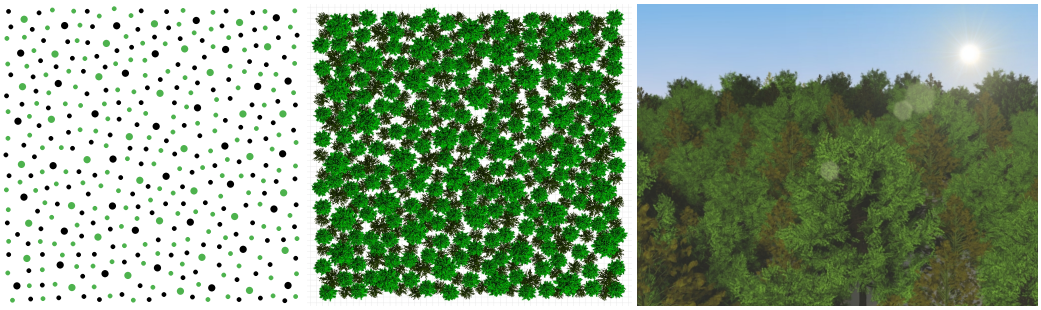

Figure 18: Distributing two kinds of trees in a forest. For each tree, two sizes were used. **Left**: Resulting distribution. **Middle**: Sampled with 3-D meshes of pines and beeches (CC-BY, Yorik von Havre). **Right**: Image rendered with "Blender".

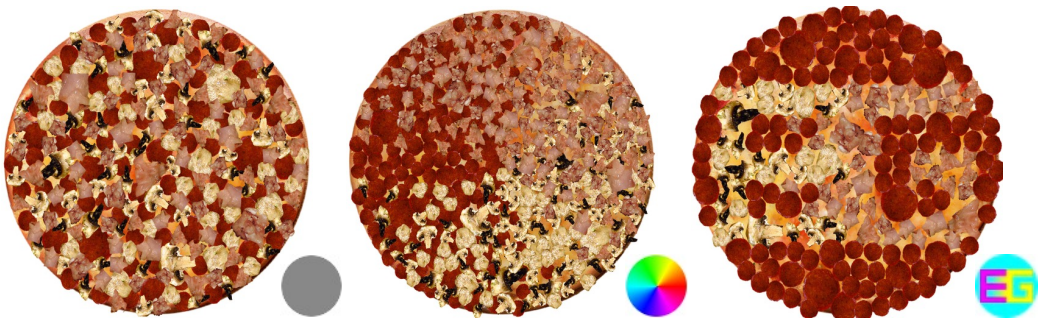

Figure 19: Pizzas whose ingredient positions are determined with our sampling approach. We put ham (User:fugutabetai_shyashin, flickr.com, CC-BY-NC-SA), mushrooms (User:cyclonebill, flickr.com, CC-BY-SA), and salami (User:vagabondvince310, flickr.com, CC-BY) on the pizzas. For each ingredient, three different sizes were used. The colours in the bottom right corners illustrate the desired density of each ingredient.

[GSWT11] GWOSDEK P., SCHMALTZ C., WEICKERT J., TEUBER T.: *Fast Electrostatic Halftoning*. Tech. Rep. 295, Department of Mathematics, Saarland University, Saarbrücken, Germany, June 2011.

[Hal60] HALTON J. H.: On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik 2*, 1 (1960), 84–90.

[HDK01] HILLER S., DEUSSEN O., KELLER A.: Tiled blue noise samples. In *VMV '01: Proc. Vision, Modeling, and Visualization* (2001), Aka, pp. 265–272.

[JEGPO02] JODOIN P.-M., EPSTEIN E., GRANGER-PICHÉ M., OSTROMOUKHOV V.: Hatching by example: a statistical approach. In *Proc. 2nd International Symposium on Non-photorealistic Animation and Rendering* (New York, NY, USA, 2002), ACM, pp. 29–36.

[KCDL06] KOPF J., COHEN-OR D., DEUSSEN O., LISCHINSKI D.: Recursive Wang tiles for real-time blue noise. *ACM Transactions on Graphics 25*, 3 (2006), 509–518.

[Kip01] KIPPHAN H.: *Handbook of print media: Technologies and production methods*. Springer, 2001.

[KK03] KOLLIG T., KELLER A.: Efficient illumination by high dynamic range images. In *EGRW '03: Proc. 14th Eurographics Workshop on Rendering* (Aire-la-Ville, Switzerland, 2003), Christensen P., Cohen-Or D., (Eds.), Eurographics Association, pp. 45–50.

[LD06] LAGAE A., DUTRÉ P.: An alternative for wang tiles: Colored edges versus colored corners. *ACM Transactions on Graphics 25*, 4 (2006), 1442–1459.

[LD08] LAGAE A., DUTRÉ P.: A comparison of methods for generating poisson disk distributions. *Computer Graphics Forum 27*, 1 (March 2008), 114–129.

[LNW*09a] LI H., NEHAB D., WEI L.-Y., SANDER P. V., FU C.-W.: *Fast Capacity Constrained Voronoi Tessellation*. Tech. Rep. MSR-TR-2009-174, Microsoft Research, 2009.

[LNW*09b] LI H., NEHAB D., WEI L.-Y., SANDER P. V., FU C.-W.: Fast capacity constrained voronoi tessellation. In *Proc. 2009 Symposium on Interactive 3D Graphics and Games* (New York, NY, USA, 2009), ACM, pp. 13:1–13:1.

[MF92] MCCOOL M., FIUME E.: Hierarchical Poisson disk sampling distributions. In *Proceedings of the Conference on Graphics Interfaces '92* (San Francisco, CA, USA, 1992), Booth K. S., Fournier A., (Eds.), Morgan Kaufmann Publishers Inc., pp. 94–105.

[NVi11] NVIDIA CORPORATION: *NVidia CUDA Programming Guide*, 4th ed., Mar. 2011. `http://developer.download.nvidia.com/compute/cuda/4_0_rc2/toolkit/docs/CUDA_C_Programming_Guide.pdf`, Retrieved June 11, 2011.

[ODJ04] OSTROMOUKHOV V., DONOHUE C., JODOIN P.-M.: Fast hierarchical importance sampling with blue noise properties. *ACM Transactions on Graphics 23*, 3 (2004), 288–495.

[PB94] PNUELI Y., BRUCKSTEIN A. M.: Digidürer - a digital engraving system. *The Visual Computer 10*, 5 (1994), 277–292.

[PH04] PHARR M., HUMPHREYS G.: *Physically Based Rendering – from Theory to Implementation*. Morgan Kaufmann, 2004.

[PHWF01] PRAUN E., HOPPE H., WEBB M., FINKELSTEIN A.: Real-time hatching. In *Proc. SIGGRAPH 2001* (2001), ACM, pp. 579–584.

[SAG03] SURAZHSKY V., ALLIEZ P., GOTSMAN C.: Isotropic remeshing of surfaces: A local parameterization approach. In *Proc. 12th International Meshing Roundtable* (2003), pp. 215–224.

[Sec02] SECORD A.: Weighted Voronoi stippling. In *Proc. Second International symposium on Non-photorealistic Animation and Rendering* (New York, NY, USA, 2002), ACM, pp. 37–43.

[SGBW10] SCHMALTZ C., GWOSDEK P., BRUHN A., WEICKERT J.: Electrostatic halftoning. *Computer Graphics Forum 29*, 8 (Dec. 2010), 2313–2327.

[SHS02] SECORD A., HEIDRICH W., STREIT L.: Fast primitive distribution for illustration. In *Proc. 13th Eurographics Workshop on Rendering* (Aire-la-Ville, Switzerland, Switzerland, 2002), EGRW '02, Eurographics Association, pp. 215–226.

[TSG*11] TEUBER T., STEIDL G., GWOSDEK P., SCHMALTZ C., WEICKERT J.: Dithering by differences of convex functions. *SIAM Journal on Imaging Sciences 4*, 1 (2011), 79–108.

[Uli87] ULICHNEY R. A.: *Digital Halftoning*. MIT Press, Cambridge, MA, 1987.

[VO08] VANDERHAEGHE D., OSTROMOUKHOV V.: Polyomino-based digital halftoning. In *IADIS International Conference on Computer Graphics and Visualization 2008* (July 2008), Isaías P., (Ed.), pp. 11–18.

[Wan61] WANG H.: Proving theorems by pattern recognition II. *Bell Systems Technical Journal* (1961), 1–42.

[Wei10] WEI L.: Multi-class blue noise sampling. *ACM Transactions on Graphics 29*, 4 (2010), Article No. 79.

[WS94] WINKENBACH G. A., SALESIN D. H.: Computer-generated pen-and-ink illustration. In *Proc. SIGGRAPH 1994* (New York, NY, USA, 1994), ACM, pp. 91–100.

[ZISS04] ZANDER J., ISENBERG T., SCHLECHTWEG S., STROTHOTTE T.: High quality hatching. *Computer Graphics Forum 23*, 3 (2004), 421–420.