We would like to thank the two anonymous referees for the time and effort they took to carefully review our manuscript and for their valuable comments and suggestions that contributed to improve its quality. Below, we give our responses (in blue) to the specific comments of the referees (in black).

Referee 1

It is a good paper. The idea of approximating Guassian weights using the central limit theorem (CLT) is very good. The Approximation 3 is a happy result mainly because of cancellation of the common factors in the numerator and the denominator. The algorithms for implementation are also nice and thorough. It is surprising to read how the computing time is reduced from a naive Barnes interpolation. Nevertheless, I have some comments and questions, basically for a better presentation.

1. It uses a special kind of the uniform random variable on an interval with a threshold. Hope the author describe why this threshold (sqrt(3/n) sigma) was chosen.

The calculation, why the interval has the length (sqrt(3/n) sigma) is actually not obvious. It is basically chosen in this way, such that the variance of the resulting uniform distribution u_n(x) is (sigma^2/n).

We added on page 3, line 65 of the manuscript a short comment about that and a calculation that verifies this.

2. page 3, line 54: hope the author provide the definition of the n-fold convolution of p(x) with itself.

We provide the definition of the n-fold convolution of p(x) with itself in the new Appendix A1 of the manuscript.

3. p5 line 92: hope the author provide more concretely what *^nx and *^yn are, because some readers may know it after reading the Algorithms 2, 3, and 4.

We fully agree, the explanation of the operators *^xn and *^yn in the text is minimalistic. We give now a thorough derivation of them in the new Appendix A2 of the manuscript

4. In Section 5, the author applied 4-fold convolution. It is known that a summation of uniform random variables converge slowly to the CLT compared to other uni-modal random variables, even though the convergence in distribution and the convergence in pdf are different. Thus I think only 4-fold is too small. I would recommend to apply at least 10-fold convolutions in real applications to ensure a good approximation.

Maybe one has to differentiate from application to application. When Barnes interpolation is used to visualize data by means of isolines - as done in the manuscript - we noticed that the resulting isolines found more or less their stable form already after applying a 3-fold convolution. For other applications, which require a higher precision, a 10-fold convolution or even more might make sense.

We extended the corresponding remark on p. 18 line 295 accordingly.

5. The result in this paper is only a nice approximation to Barnes interpolation. I think the title can be changed to 'Fast approximation to Barnes interpolation'.

We did not use the term "approximate" in the title, because other _feasible_ methods to compute Barnes interpolation are approximations as well, but do not emphasize this in an obvious way.

The title of the manuscript is now “Fast Approximate Barnes Interpolation”.

6. It uses a special kind of the uniform random variable and a numerical integration for convolution in Algorithm 2. It is still good, but I guess one may consider other random variables which may accelerate the convergene to the CLT. Then numerical integration may be a little more complex than that in this paper. This may increase computing time in Algorithm 2, but may be alright with a smaller n.

In fact, there are many other PDFs that have a faster CLT-convergence to a Gaussian than the uniform PDF - in the extreme case one could even take a normal distribution itself for which we would have n = 1.

But when using a non-trivial PDF, it is clear that putting and moving the weight window (as described on p. 9 line 161) over the data to be convolved requires in general 2T+1 multiplications and 2T additions per element. Thus, the simplicity of algorithm 2 is lost to a far part and the algorithmic complexity of it grows to O(L*T) instead of O(L). Overall we then could not claim a complexity of O(N+W*H) anymore, this would be rather O(N+W*H*T).

Your comment leads us to consider to add a remark on p. 6 line 128 that Approximation 3 is valid in a much more general context, i.e. also for other (non-uniform) PDFs. In the case of a normal distribution even equality holds. These relationships are now described in the new Appendix A3 of the manuscript.

7. In Algorithms 2 and 3, I was a bit confused with the notations g, r_T, and F[i,j]. Hope the author kindly give short descriptions on these notations to improve reader's understanding.

These notations basically refer to the formulae and text given around the Algorithms.

We addressed this problem in chapter 4 Discretization by first adding a hint about the notation that we use round parentheses for functions g(x) defined on the continuum and square brackets for discrete functions g[x]. Throughout the chapter, we carefully improved the readability by giving more context here and there and by being more specific.

8. Finally, I think that the idea of approximating Guassian weights using the CLT is very good and so it can be applied to the other area of numerical computations which use Gaussian weights. Hope the author try to search the literature if any body has already published or applied this idea.

This is a good question. The CLT is used by some applications to quickly generate the weights of a normal distribution (as above, of course only in an approximate way). Further we assume that some image processing programs like Photoshop employ this technique in order to apply a Gaussian blur filter on an image. This process can be regarded as a special case of the method presented in this manuscript, where the observation points in question are given by the image matrix. In this case the points are located on a regular grid and the denominator of equation (8) simplifies to 1.

We refer in the last paragraph of the Introduction to applications of the CLT in image processing and computer vision (Wells, 1986). Further we refer in chapter 5.4 on p. 16, line 269 and on p. 17, line 283 to (Gwosdek et al., 2011), where in principle the same considerations are made for so called “extended box filters”. Finally, we mention in the Conclusions (Getreuer, 2013), who investigated different “Gaussian convolution algorithms”.

The References were updated with (Getreuer, 2013), (Gwosdek et al., 2011) and (Wells, 1986).

Referee 2

This is a nice paper. It presents an efficient method to perform a Gaussian smoothing, which takes advantage of the CLT by using the repeated convolution of a computationally cheap (uniform) smoothing to approximate the desired Gaussian result (and further that the 2D result is the product of two 1D calculations).

I see that the author already improved the first version of the manuscript in response to previous comments, and I found it very readable. I found that the questions that arose on my first reading were answered later on. I have a minor suggestion about the presentation of the computational complexity of the main result, which is that the wording could be more precise. For example, in the abstract "When implemented naively, the computational complexity of Barnes interpolation depends directly on both the number of sample points and the number of grid points." This initially confused me as all algorithms (including the proposed one as well as the naive approach) must surely depend directly on these numbers, simply from the time taken to read and write them. It would be better expressed as varying with the *product* of the number of sample points and grid points. Or else (additionally?) write it in O() notation so there can be no confusion. The Abstract should also explicitly present the complexity of the new result (and perhaps both old and new complexities could be placed again in the Conclusions).

We agree that this can be written more precise and also should be written with more emphasis, as it is ultimately the main achievement of the presented manuscript. Hence, the Abstract now contains the computational complexity in explicit O() notation for both, the naive approach and the newly presented method.

As you suggested, the first paragraph of the Conclusions now also contrasts the computational complexity of the naive approach and the presented approach.

Also one piece of ungrammatical English at l133: "are sufficiently good contained" could be "sufficiently well contained" perhaps?

This is corrected, thank you for the hint.

I suggest that the Conclusion contain a repetition of the author's suggestion that 4 convolutions seems to work well in most practical applications.

We agree with this. This suggestion is now also included in the first paragraph of the Conclusions.