**FRIEDRICH-ALEXANDER-UNIVERSITÄT ERLANGEN-NÜRNBERG**
INSTITUT FÜR INFORMATIK (MATHEMATISCHE MASCHINEN UND DATENVERARBEITUNG)

**Lehrstuhl für Informatik 10 (Systemsimulation)**

# Kaczmarz Extended Algorithm for Tomographical Image Reconstruction from Limited-Data

Constantin Popa and Rafal Zdunek

Lehrstuhlbericht 03-6

**Constantin Popa** visiting professor (DAAD Gastdozent) at Friedrich-Alexander-University
Erlangen-Nuremberg, Germany
on leave from "Ovidius" University of Constanţa, Romania
cpopa@immd10.informatik.uni-erlangen.de

**Rafal Zdunek** Insitute of Telecommunications and Acoustics,
Wroclaw University of Technology,
Wybrzeze Wyspianskiego 27, 50–370 Wroclaw, Poland
rafzdu@zstux.ita.pwr.wroc.pl

**Abstract**

Computerized Tomography (CT) was introduced in the early 1970s for diagnostic radiology, but since then many other applications of it have developed and become known (e.g. electromagnetic geotomography). CT is concerned with reconstruction of a function from its line or plane integrals, such a kind of problems being known as "inversion problems". An important class of inversion problems are concerned with image reconstruction from projections, for which the Algebraic Reconstruction Technique (ART), based on the well known algorithm proposed by S. Kaczmarz in 1937, is one of the most important class of solution methods. But unfortunately, almost all the methods from the ART class give satisfactory results only in the case of consistent problems. In the inconsistent case (and unfortunately this is what happens in real applications, because of measurement errors) they give only more or less "controllable" versions of the exact solutions. This is exactly the case that we analyze in the present paper. We start with a theoretical analysis of the classical Kaczmarz's projection method in the case of an inconsistent linear least-squares problem and we prove that the approximations so obtained are at a certain distance from the set of exact least-squares solutions. This distance is controlled by the component of the right hand side of the problem lying in the orthogonal complement of the range of problem's matrix, i.e. exactly the component that makes the problem inconsistent. For overcoming this difficulty we consider an extended version of Kaczmarz's algorithm, previously analyzed by one of the authors. In the numerical experiments described in the last part of the paper we compare the above mentioned extension with two well known (ART) type algorithms for image reconstruction in two electromagnetic geotomography problems. The results indicate that the extended Kaczmarz algorithm gives much better results than the other two.

**AMS Subject Classification:** 65F10, 65F20

**Key words and phrases:** linear least-squares problems, Kaczmarz algorithm, limited-data tomography, EM geotomography

# 1 Kaczmarz's algorithm for inconsistent problems

We consider the linear least-squares problem: find $x^* \in I\!R^n$ such that

$$\| Ax^* - b \| = \min!, \tag{1}$$

where $A$ is an $m \times n$ real matrix and $b \in I\!R^m$ a given vector (by $\| \cdot \|$ and $\langle \cdot, \cdot \rangle$ we shall denote the Euclidean norm and scalar product on some space $I\!R^q$). Let $LSS(A, b)$ be the set of all its least-squares solutions and $x_{LS}$ its minimal norm one. Also $B^t, R(B), N(B)$ will denote the transpose, range and null space of a matrix $B$, respectively, and $P_S$ the orthogonal projection onto a closed convex set $S \subset I\!R^q$. By $a_i \in I\!R^n$, we shall denote the $i$-th row of $A$ and we shall suppose that

$$a_i \neq 0, \ \forall i = 1, \ldots, m. \tag{2}$$

Moreover, all the vectors that will appear will be considered as column vectors. We have the equality (see e.g. [1])

$$LSS(A; b) = x_{LS} + N(A). \tag{3}$$

Moreover, with respect to the decomposition of $b$ as

$$b = b_A + b_A^*, \ b_A = P_{R(A)}(b), \ b_A^* = P_{N(A^t)}(b) \tag{4}$$

we have the equivalent formulation of (1)

$$x^* \in LSS(A;b) \Leftrightarrow Ax^* = b_A, \tag{5}$$

together with that given by the normal equation

$$x^* \in LSS(A;b) \Leftrightarrow A^t Ax^* = A^t b. \tag{6}$$

Let $f_i(b;\cdot), F(b;\cdot) : I\!R^n \longrightarrow I\!R^n$ be the applications defined by

$$f_i(b;x) = x - \frac{\langle x, a_i\rangle - b_i}{\| a_i \|^2} a_i,\ i = 1, \ldots, m, \tag{7}$$

$$F(b;x) = (f_1 \circ \ldots \circ f_m)(b;x), \tag{8}$$

**Kaczmarz's algorithm (K)**. Let $x^0 \in I\!R^n$; for $k = 0, 1, 2, \ldots$ do

$$x^{k+1} = F(b;x^k). \tag{9}$$

The following convergence result was proved in [17].

**Theorem 1** *(i) The sequence $(x^k)_{k\geq 0}$ from (9) always converges and*

$$\lim_{k\to\infty} x^k = P_{N(A)}(x^0) + Gb, \tag{10}$$

*where $G$ is an $n \times m$ generalized inverse of $A$ (see (16) below).*
*(ii) If the problem (1) is consistent, i.e.*

$$b \in R(A) \Leftrightarrow b = b_A, \tag{11}$$

*then the limit in (10) is a solution of (1). Moreover, we have the characterizations*

$$LSS(A;b) = \{P_{N(A)}(x^0) + Gb, x^0 \in I\!R^n\} = Gb + N(A), \quad x_{LS} = Gb. \tag{12}$$

**Remark 1** *In the consistent case (11), the limit in (10) is independent of the ordering of the rows $a_i$ inside Kaczmarz's algorithm (see (8)), i.e. for any other ordering we have the same limit from (10), which is a solution of the problem and depends only on the initial approximation $x^0$.*

**Remark 2** *Unfortunately, in the inconsistent case, i.e.*

$$b \notin R(A) \Leftrightarrow b_A^* \neq 0, \tag{13}$$

*the limit in (10) is no more a least-squares solution of (1). We shall analyze this aspect in what follows.*

**Remark 3** *In the inconsistent case (13), the limit in (10) also depends on the ordering of the rows $a_i$. We are not interested on this aspect in the present paper and for details we send the reader to [6] and references therein. But, this dependence on the row ordering means that we can no more have a clear control about the results obtained with the classical Kaczmarz algorithm (9).*

We shall now start the theoretical analysis mentioned in Remark 2 before. More clear, in the inconsistent case (13), denoting by $x^*(x^0)$ the limit in (10), i.e.

$$x^*(x^0) = P_{N(A)}(x^0) + Gb \tag{14}$$

we shall analyze some properties of $x^*(x^0)$ in connection with the problem (1) and the set of its least-squares solutions $LSS(A;b)$. In this sense we need to introduce the following matrices (see also [17])

$$P_i = I - \frac{1}{\| a_i \|^2} a_i a_i^t, \quad P_i x = x - \frac{\langle x, a_i\rangle}{\| a_i \|^2} a_i, \tag{15}$$

$$Q = P_1 \ldots P_m, \quad \tilde{Q} = QP_{R(A^t)}, \quad G = (I - \tilde{Q})^{-1} R, \tag{16}$$

$$R = col\,(R_1, \ldots, R_m), \quad R_i = \frac{1}{\| a_i \|^2} P_1 \ldots P_{i-1} a_i. \tag{17}$$

**Remark 4** *In [17] it is proved that*

$$\| \tilde{Q} \|_2 = \sup\{\| Qx \|, x \in R(A^t), \| x \| \leq 1\} < 1 \tag{18}$$

*(where by* $\| \cdot \|_2$ *we denoted the spectral norm of a matrix) and that the matrix* $G$ *from (16) satisfies*

$$AGA = A,\ GAG = G,\ (GA)^t = GA. \tag{19}$$

**Lemma 1** *We have the equalities*

$$LSS(A;b) = Gb_A + N(A), \quad x_{LS} = Gb_A \in R(A^t). \tag{20}$$

**Proof.** Let $S(A;b_A)$ be the set of all solutions of the system $Ax = b_A$. From (5) it results

$$LSS(A;b) = S(A;b_A). \tag{21}$$

But, from (11)-(12) we get

$$S(A;b_A) = LSS(A;b_A) = Gb_A + N(A),$$

which combined with (21) gives us the first equality in (20). The second one holds from (11) and (12).

**Proposition 1** *The vector* $x^*(x^0)$ *from (14) is a least-squares solution of the (right hand side) perturbed problem*

$$\| Az^* - (b_A + AGb_A^*) \| = \min! \tag{22}$$

*with* $b_A, b_A^*$ *from (4) and* $G$ *from (16).*

**Proof.** From (4), (20) and (21) we successively get

$$A^t Ax^*(x^0) = A^t A(P_{N(A)}(x^0) + Gb) = A^t AGb =$$

$$A^t A(Gb_A + Gb_A^*) = A^t(Ax_{LS} + AGb_A^*) = A^t(b_A + AGb_A^*). \tag{23}$$

But (23) is exactly the normal equation associated to (22) and the proof is complete.
Let now $d(v,w)$ be the Euclidean distance between the vectors $v, w \in \mathbb{R}^n$ defined by

$$d(v,w) = \| v - w \| . \tag{24}$$

We have the following result

**Proposition 2** *The distance between the element* $x^*(x^0)$ *from (14) and the set* $LSS(A;b)$ *satisfies the following inequality*

$$d\left(x^*(x^0), LSS(A;b)\right) = \inf_{z^* \in LSS(A;b)} \| x^*(x^0) - z^* \| \quad \leq$$

$$\frac{\delta}{1 - \| \tilde{Q} \|_2} \left( \sum_{i=1}^{m} \frac{1}{\| a_i \|^2} \right)^{1/2}, \tag{25}$$

*where* $\delta$ *is a positive real number satisfying*

$$\| b_A^* \| \leq \delta. \tag{26}$$

**Proof.** From (20), (14), (4) and (16) we obtain

$$d\left(x^*(x^0), LSS(A;b)\right) = \inf_{z^0 \in \mathbb{R}^n} \| x^*(x^0) - (P_{N(A)}(z^0) + Gb_A) \| \leq$$

$$\| x^*(x^0) - (P_{N(A)}(x^0) + Gb_A) \| = \| Gb - Gb_A \| =$$
$$\| Gb_A^* \| \quad \leq \quad \| (I - \tilde{Q})^{-1} \|_2 \quad \| R \|_2 \quad \| b_A^* \| . \tag{27}$$

From (18) we get (see also [8])

$$\| (I-\tilde{Q})^{-1} \|_2 \ \leq \ \frac{1}{1- \| \tilde{Q} \|_2}. \tag{28}$$

For $\| R \|_2$, from (17) we get (for an arbitrary $y \in I\!R^m$)

$$\| Ry \| = \| \sum_{i=1}^{m} y_i R_i \| \leq \sum_{i=1}^{m} |y_i| \, \| R_i \| \ \leq$$

$$\left( \sum_{i=1}^{m} |y_i|^2 \right)^{1/2} \left( \sum_{i=1}^{m} \| R_i \|^2 \right)^{1/2} = \ \| y \| \left( \sum_{i=1}^{m} \| R_i \|^2 \right)^{1/2}. \tag{29}$$

But, using the fact that $\| P_i \|_2 = 1, \forall i = 1, \ldots, m$ (see [17]) we obtain for the vectors $R_i$ from (17)

$$\| R_i \| = \frac{1}{\| a_i \|^2} \| P_1 \ldots P_{i-1} a_i \| \leq \frac{1}{\| a_i \|^2} \| a_i \| \ = \ \frac{1}{\| a_i \|}. \tag{30}$$

Then, from (29)-(30) we get

$$\| R \|_2 = \sup_{y \neq 0} \frac{\| Ry \|}{\| y \|} \leq \left( \sum_{i=1}^{m} \frac{1}{\| a_i \|^2} \right)^{1/2}. \tag{31}$$

By combining the above inequalities with (27) we get (25) and the proof is complete.

**Remark 5** *The upper bound in (25) tells us that in evaluating the distance between $x^*(x^0)$ and $LSS(A;b)$ an important role plays the vector $b_A^*$ from (4) (see also (26)) which can be considered as a "perturbation from consistency" in problem (1). This very important aspect will become more clear according to the next result.*

**Proposition 3** *The following equality is true*

$$d\left(x^*(x^0), LSS(A;b)\right) \geq \| Gb_A^* \| . \tag{32}$$

**Proof.** Let $LK(A;b)$ be the set of all limit points in (10), i.e.

$$LK(A;b) = \{P_{N(A)}(x^0) + Gb, x^0 \in N(A)\} = Gb + N(A) \tag{33}$$

(note that in the consistent case for (1) we have $LK(A;b) = LSS(A;b)$). Let $\hat{x}_{LS}$ be the minimal norm solution of the perturbed problem (22). Because the right hand side of (22) belongs to the space $R(A)$, from (11)-(12) (also using (19) and (4)) it results that

$$\hat{x}_{LS} = G(b_A + AGb_A^*) = Gb_A + GAGb_A^* = Gb_A + Gb_A^* = Gb \tag{34}$$

and

$$LSS(A; b_A + AGb_A^*) = Gb + N(A) = LK(A;b). \tag{35}$$

From (34)-(35) and (20) we obtain that both sets $LSS(A;b)$ and $LK(A;b)$ are translations of $N(A)$ by vectors from $R(A^t)$, which is its orthogonal complement. This means that we have

$$d\left(LSS(A;b), LK(A;b)\right) = \inf\{d(u,v), u \in LSS(A;b), v \in LK(A;b)\} =$$

$$d(x_{LS}, \hat{x}_{LS}) = \| Gb_A - Gb \| = \| Gb_A^* \| . \tag{36}$$

Then, (32) follows from the fact that $x^*(x^0) \in LK(A;b)$, thus

$$d\left(x^*(x^0), LSS(A;b)\right) \geq d\left(LK(A;b), LSS(A;b)\right) \tag{37}$$

and the proof is complete.

**Corollary 1** *For any limit point $x^*(x^0)$ in (10) we have the equality*

$$d\left(x^*(x^0), LSS(A;b)\right) = \| Gb_A^* \| . \tag{38}$$

**Proof.** It results from (32) and (27).

**Remark 6** *All the above considerations have been made with respect to the row ordering from (8) inside the Kaczmarz's iteration. If we change it, the matrices $Q, \tilde{Q}$ and $R$ from (16)-(17) will change, and so will do $G$, but all the results rest true with respect to these new ones.*

## 2 Kaczmarz's algorithm with relaxation parameters

The results from (25) and (38) in the previous section tell us that, when applying classical Kaczmarz algorithm (7)-(8) to an inconsistent problem (1) for which the "perturbation from consistency" $b_A^*$ is big (see also Remark 5), the obtained approximation $x^*(x^0)$ can be enough far from the solution set. This can explain the troubles appearing in image reconstruction problems for which (because of measurement errors) the right hand side $b$ changes and (1) becomes inconsistent (see also the comments in [3], [5], [1], [12]). One possibility to overcome this difficulty is to introduce relaxation parameters in the iteration (7)-(8). A first approach for this, that we shall consider here was described in [12]. It replaces the projections in (7)-(8) by

$$f_i(\omega; b; x) = (1-\omega)x + \omega f_i(b; x),\ i = 1, \ldots, m, \tag{39}$$

$$F(\omega; b; x) = (f_1 \circ \ldots \circ f_m)(\omega; b; x), \tag{40}$$

for $\omega \neq 0$ a given real number. We then obtain the following method (**K**aczmarz with **R**elaxation parameters).
**Algorithm KR.** Let $x^0 \in I\!R^n$; for $k = 0, 1, \ldots$ do

$$x^{k+1} = F(\omega; b; x^k). \tag{41}$$

In [12] it is proved that, for any $x^0 \in I\!R^n$ and $\omega \in (0, 2)$, the sequence $(x^k)_{k\geq 0}$ from (41) converges to a vector $\tilde{x}^*(x^0; \omega)$. Moreover, we have

$$\lim_{\omega \to 0} \tilde{x}^*(x^0; \omega) = \tilde{x}^*(x^0), \tag{42}$$

where $\tilde{x}^*(x^0)$ is a least-squares solution of the following weighted formulation of (1)

$$\| D^{-1}(Ax^* - b) \| = \min! \tag{43}$$

where

$$D = \mathrm{diag}(\| a_1 \|, \ldots, \| a_m \|). \tag{44}$$

**Remark 7** *We have to mention here that, by taking into account the results from [15] all the consideration and results from section 1 rest true also for the algorithm* **KR***, with corresponding matrices (as in (15)- (17), but depending also on* $\omega$*).*

Another approach was described in [3] (see also [10] for a generalization of it). It considers, instead of the applications $f_i(\omega; b; x), F(\omega; b; x)$ from (39)-(40), $g_i(\lambda; b; x)$ and $\Gamma(\lambda; b; x)$ given by

$$g_i(\lambda; b; x) = x + \lambda(b_i - \langle x, a_i \rangle)a_i, i = 1, \ldots, m, \tag{45}$$

$$\Gamma(\lambda; b; x) = (g_1 \circ \ldots \circ g_m)(\lambda; b; x). \tag{46}$$

We then get the following algorithm (**C**ensor, **E**ggermont, **G**ordon).
**Algorithm CEG.** Let $x^0 \in I\!R^n$; for $k = 0, 1, \ldots$ do

$$x^{k+1} = \Gamma(\lambda; b; x^k). \tag{47}$$

In [3] it is proved that, for any $x^0 \in I\!R^n$ and $\lambda$ satisfying

$$\lambda \in \left(0, \min_{1 \leq i \leq m} \frac{2}{\| a_i \|^2}\right) \tag{48}$$

the sequence $(x^k)_{k\geq 0}$ from (47) converges to a vector $\hat{x}^*(x^0; \lambda)$. Moreover, we have

$$\lim_{\lambda \to 0} \hat{x}^*(x^0; \lambda) = \hat{x}^*(x^0), \tag{49}$$

where $\hat{x}^*(x^0)$ is a least-squares solution of the weighted formulation (43)-(44).

**Remark 8** *Concerning the two above Kaczmarz-like algorithms, we have to firstly observe that the use of very small values of the relaxation parameters $\omega$ and $\lambda$ in (41) and (47), respectively will (sometimes dramatically) slow the convergence of the corresponding sequence of approximations.*

But, probably the most unpleasant aspect related to the above algorithms is the fact that, although we consider the limit value in (42) or (49), in the inconsistent case for (1) it represents a solution of the weighted problem (43)-(44) and not of the original one (1). Thus, essential differences can appear between the corresponding reconstructed images. A theoretical information about this can be obtained if we consider a particular case of the analysis and results from [9]. For this, let

$$M = I,\ \tilde{M} = D^2,\ \Delta M = \tilde{M} - M,\ \tilde{A} = A,\ \tilde{b} = b \tag{50}$$

as in [9] (with $D$ from (44)). Then, our problems (1) and (43) will become

$$\langle M^{-1}(Ax^* - b), Ax^* - b\rangle = \min! \tag{51}$$

and, respectively

$$\langle \tilde{M}^{-1}(Ax^* - b), Ax^* - b\rangle = \min! \tag{52}$$

Let us firstly observe that (52) can be also considered as a "non-weighted" least-squares problem, but with respect to the scalar product (on $\mathbb{R}^n$) defined by

$$\langle x, y\rangle_{\tilde{M}} = \langle \tilde{M}^{-1}x, y\rangle = \langle D^{-2}x, y\rangle. \tag{53}$$

Let $x_{LS}$ and $\tilde{x}_{LS}$ be the minimal norm solution of (51) and (52), respectively (i.e. of (1) and (43), respectively). Then, by taking

$$\tilde{\lambda} = P^{\tilde{M}}_{N(\tilde{A}^t)}(b), \tag{54}$$

where $\tilde{A}^t$ is the transpose of $A$ with respect to the scalar product from (53) and $P^{\tilde{M}}_{N(\tilde{A}^t)}(b)$ is the orthogonal projection of $b$ onto the subspace $N(\tilde{A}^t)$, but with respect to the same scalar product in (53), and by following (50) and the estimation (3.5) in [9] we get

$$\tilde{x}_{LS} - x_{LS} = A^+(D^2 - I)\tilde{\lambda}. \tag{55}$$

The above equality tells us that the difference between the two minimal norm solutions essentially depends on the vector $\tilde{\lambda}$ from (54) which, similarly as $b^*_A$ (see (4), (26) and (38)) measures the "perturbation from consistency" in the problem (52) (with respect to the scalar product (53)). Thus, "bigger" is $\tilde{\lambda}$, bigger will be the distance in (55), thus the weighted solution $\tilde{x}_{LS}$ can "spoil" the reconstructed image.

# 3 The extended Kaczmarz algorithm

In [14] (see also [13], [15]) one of the authors proposed a different extension of Kaczmarz's algorithm (7)-(9) for dealing with the inconsistent case for (1). For this, we have to consider, beside the orthogonal projections $f_i(b; x)$ from (7) onto the hyperplanes generated by the rows $a_i$ of $A$, similar ones but with respect to the columns, denoted here by $\alpha_j \neq 0, j = 1, \ldots, n$. They are defined by

$$\varphi_j(y) = y - \frac{\langle y, \alpha_j\rangle}{\| \alpha_j \|^2}\alpha_j,\ j = 1, \ldots, n, \tag{56}$$

$$\Phi(y) = (\varphi_1 \circ \ldots \circ \varphi_n)(y). \tag{57}$$

Then the **E**xtended **K**aczmarz algorithm is defined as follows.
**Algorithm KE.** Let $x^0 \in \mathbb{R}^n, y^0 = b$; for $k = 0, 1 \ldots$ do

$$y^{k+1} = \Phi(y^k), \tag{58}$$

$$b^{k+1} = b - y^{k+1}, \tag{59}$$

$$x^{k+1} = F(b^{k+1}; x^k). \tag{60}$$

**Remark 9** *In the above step (60) the components $b_i$ of the right hand side $b$ are replaced in (7)-(8), at each iteration by the corresponding components $b_i^{k+1}$ of the vector $b^{k+1}$ from (59). Moreover, we can observe that from a computational complexity point of view, the algorithm* **KE** *is only "a constant times" per iteration more expensive than the original Kaczmarz's one (8) (this constant is related to the ratio $\frac{m}{n}$ and the sparsity of $A$).*

In [14] it was proved that, for any initial approximation $x^0$, the algorithm **KE** generates a convergent sequence $(x^k)_{k\geq 0}$ and

$$\lim_{k\to\infty} x^k = P_{N(A)}(x^0) + Gb_A. \tag{61}$$

Thus, according to (20) the limit vector in (61) is always a least-squares solution of (1) and the minimal norm one $x_{LS}$ is obtained if and only if we start with $x^0 \in R(A^t)$ (e.g. $x^0 = 0$). Moreover, we can construct a version of **KE** which includes relaxation parameters in the steps (58) and (60). For this, we consider instead of the applications from (56)-(57), $\varphi_j(\alpha; y)$ and $\Phi(\alpha; y)$ defined by

$$\varphi_j(\alpha; y) = (1-\alpha)y + \alpha\varphi_j(y), j = 1, \ldots, n, \tag{62}$$

$$\Phi(\alpha; y) = (\varphi_1 \circ \ldots \circ \varphi_n)(\alpha; y). \tag{63}$$

Then, the **K**aczmarz **E**xtended with **R**elaxation **P**arameters algorithm is the following.
**Algorithm KERP.** Let $x^0 \in I\!R^n, y^0 = b$; for $k = 0, 1 \ldots$ do

$$y^{k+1} = \Phi(\alpha; y^k), \tag{64}$$

$$b^{k+1} = b - y^{k+1}, \tag{65}$$

$$x^{k+1} = F(\omega; b^{k+1}; x^k). \tag{66}$$

with $F(\omega; b; x)$ from (39)-(40). In [15] it was proved that, for any $x^0 \in I\!R^n$ and any $\omega, \alpha \in (0, 2)$ the sequence $(x^k)_{k\geq 0}$ generated by **KERP** algorithm converges always to a least-squares solution of (1).

**Remark 10** *Block-versions and other developments of the above algorithms* **KE** *and* **KERP** *were presented in [15], [16].*

# 4 Experiments

## 4.1 Right hand side perturbation

As we have already mentioned at the beginning of section 2, in practical applications perturbations can appear in the right hand side of (1) because of measurement errors in the process of scanning the given object. Let us suppose that the (initial) problem (1) is consistent, i.e. (11) holds. The perturbed one will be

$$\| Ax^* - \tilde{b} \| = \min!, \tag{67}$$

where

$$\tilde{b} = b + \delta b,\ \delta b = \delta b_A + \delta b_A^*,\ \delta b_A \in R(A),\ \delta b_A^* \in N(A^t). \tag{68}$$

According to the considerations and results from section 1, if $x^0$ is the initial approximation, the limit of the Kaczmarz sequence $(x^k)_{k\geq 0}$ from (9) applied to (67) will be given by (see (14))

$$x^*(x^0) = P_{N(A)}(x^0) + G\tilde{b} \tag{69}$$

and the distance between $x^*(x^0)$ and the set $LSS(A; b)$ by (see (38), (67), (68))

$$d\left(x^*(x^0), LSS(A; b)\right) = \| G\delta b \| = \| G\delta b_A + G\delta b_A^* \| . \tag{70}$$

Let now $z^*(x^0)$ be the limit of the Kaczmarz Extended sequence from (58)-(60) applied to (67). According to (61) we have

$$z^*(x^0) = P_{N(A)}(x^0) + GP_{R(A)}(\tilde{b}) \tag{71}$$

and a similar analysis as in section 1 gives us (by also using (11) and (68))

$$d\left(z^*(x^0), LSS(A;b)\right) = \| GP_{R(A)}(\tilde{b}) - GP_{R(A)}(b) \| =$$

$$\| Gb_A + G\delta b_A - Gb_A \| = \| G\delta b_A \| . \tag{72}$$

From (70) and (72) it results that, if the perturbation (68) "acts" more on the direction of $N(A^t)$, i.e. $\delta b_A^*$ is "much bigger" than $\delta b_A$ then the distance in (70) will be bigger than that in (72). We then expect that the corresponding reconstructed image using $z^*(x^0)$ be (much) better than that obtained with $x^*(x^0)$ from (69). A more clear idea in this sense is given by the "extreme" case in which $\delta b_A = 0$. Then $z^*(x^0) \in LSS(A;b)$, but $x^*(x^0) \notin LSS(A;b)$ and (see (70))

$$d\left(x^*(x^0), LSS(A;b)\right) = \| G\delta b_A^* \| \ > \ 0. \tag{73}$$

These theoretical considerations are confirmed by the experiments described in section 4.4.

## 4.2 Tomographical image reconstruction from limited-data

Image reconstruction from limited-data forms a group of tomographical inversions in which an angular range of the projections (rays) is limited. This is the case, for example, in the tomographical borehole technique. An example of such a technique is electromagnetic (EM) geotomography [7]. Synthetic data from this type of tomography were used in our experiments.

The EM geotomography technique maps the rock structure of an earth cross-borehole section using remote sensing measurements. The structure is represented by an image of the attenuation coefficient distribution of electromagnetic waves. The data are obtained by measuring the attenuation of the electromagnetic field strength along assumed straight-line rays which can be created between a field source moving up and down along one borehole and observation points regularly spaced along the other borehole.

Similarly as in many tomographical imaging techniques [2, 7, 11], each datum $b_i$ $(i = 1, \ldots, M)$ can be modelled as the line integral of the unknown attenuation coefficient along the path of the $i$-th ray. In a discrete representation of the image and for a fine number of projections, the line integral can be approximated by a finite sum:

$$\sum_{j=1}^{N} a_{ij} x_j = b_i \quad \text{or} \quad Ax = b, \quad A = [a_{ij}] \in I\!R^{M \times N}, \quad M \geq N \tag{74}$$

where $N$ is a number of pixels in the image, $M$ is a number of projections, $x_j$ of image vector $x = [x_1, \ldots, x_N]^T \in I\!R^N$ is the attenuation coefficient of the $j$-th pixel, $b_i$ of data vector $b = [b_1, \ldots, b_M]^T \in I\!R^M$ is a total attenuation measured along the $i$-th ray, and $a_{ij}$ represents the contribution of the $j$-th pixel to datum $b_i$.

Unfortunately, the desired solution to (74) is much more difficult to be recovered than, e.g. in Computerized Tomography (CT), mainly due to unfavorable properties of discrete operator $A$. In limited-data tomography, $A$ is rank-deficient and ill-conditioned. Thus, $N(A)$ is non-trivial. Moreover, inevitable measurement errors make system (74) inconsistent. Thus, the image reconstruction problem can be boiled down to perturbed linear least-squares problem (67).

## 4.3 Test data

The above-mentioned algorithms were implemented in MATLAB 6.1, with using vectorization as much as possible. The experiments are performed on the basis of synthetic data generated from two image models of different sizes and features. The models can be considered as a "phantom" which is commonly used for testing algorithms in medical tomography [11].

The first model, which is denoted as `Case 1`, refers to the original image shown in Figure 1 (left). The image is associated with an area of 12 by 12 meters which is sampled on the low resolution mesh - $12 \times 12$ pixels. Thus a single pixel is of the size 1 by 1 meter, and $N = 144$. The area is probed along multiple straight-line rays which are traced between 12 transmitter points regularly spaced along one borehole and 12 receiver points regularly spaced along the other borehole. Thus there is 1 point in each pixel adjacent to the borehole. The total number of rays is 144. The coefficient matrix associated with such an area is denoted by $A_1$, where $A_1 \in I\!R^{144 \times 144}$ and $0 \leq a_{ij} \leq \sqrt{2}$.

This original image contains sharp-edge inhomogeneities which make it possible to observe the interactions between adjacent pixels in the reconstructed image.

The original image of the second model, which is referred to as `Case 2`, is presented in Figure 2 (left). The area of 30 by 30 meters, associated with the original image, is sampled on the mesh of $30\times30$ pixels. Thus a total number of pixels is 900. For the same pattern of a rays-coverage as above, we have $M = 900$ and the associated coefficient matrix $A_2 \in I\!R^{900\times900}$. Such an original image imitates real objects in EM geotomography. Assuming our measurements are taken in a region of an intensive mining exploration, we can expect to have the rock structure strongly inhomogeneous. Such an area can include various cracks and air voids. Since attenuation of an electromagnetic wave in air is much lower than in earth, the original images contain sub-areas of low-value (blue color), which model the air voids. The clusters of high-value pixels (red color) represent low humid sub-areas.

The fundamental properties of the coefficient matrices are shown in Table 1. The rank of $A$ is evaluated by a standard procedure built in MATLAB 6.1. The condition number of $A$ is computed as follows: $\kappa(A) = \sigma_1/\sigma_r$, where $\sigma_1$ is the largest singular value and $\sigma_r$ is the singular value at index $r = \text{rank}(A)$.

Table 1: Properties of matrices $A_1$ and $A_2$

.

| properties | $A_1$ | $A_2$ |
|---|---|---|
| size | $144 \times 144$ | $900 \times 900$ |
| sparseness | 90.1% | 95.7% |
| rank | 125 | 805 |
| condition number | $9.3952 \times 10^4$ | $2.1548 \times 10^7$ |

Each original image after having been sampled forms image matrix $X_{exact}$. For `Case 1`, $X^{(1)}_{exact} \in I\!R^{12\times12}$, and for `Case 2`, $X^{(2)}_{exact} \in I\!R^{30\times30}$. Then, each image matrix is converted to image vector $x_{exact}$ in this way that each row of the corresponding $X_{exact}$ is stacked up one after the other. Noise-free data are computed as follows: $b = Ax_{exact}$, with reference to each case.

In order to analyze the perturbed problem in (67) and to have a full control over noisy perturbation, $\delta b$ in (68) is generated using applications (56), (57), where $y^0 = [1, 1, \ldots, 1]^T \in I\!R^M$. Sequence $y^{k+1} = \Phi(y^k)$ for $k$ large enough stabilizes to value $\tilde{y}$. For a very large number of $k$, it holds $\tilde{y} = P_{N(A^T)}(y^0) \in N(A^T)$ and the noise has the Gaussian distribution. However, we set $k$ in this way to have $||\tilde{y}_A|| \cong ||\tilde{y}^*_A||$, where $\tilde{y}_A = P_{R(A)}(\tilde{y})$ and $\tilde{y}^*_A = P_{N(A^T)}(\tilde{y})$. The projections are defined as follows:

$$P_{R(A)}(\tilde{y}) = U_{range}U^T_{range}(\tilde{y}), \qquad P_{N(A^T)}(\tilde{y}) = U_{null}U^T_{null}(\tilde{y})$$

where $U_{range} = [u_1, \ldots, u_r]$ and $U_{null} = [u_{r+1}, \ldots, u_M]$ are formed from respective right singular vectors of $A$.

Perturbations $\delta b_A$ and $\delta b^*_A$ in (68) are stated as follows:

$$\delta b_A = c_A P_{R(A)}(\tilde{y}), \qquad \delta b^*_A = c^*_A P_{N(A^T)}(\tilde{y})$$

where constants $c_A$ and $c^*_A$ determine the appropriate amount of noise.

We analyze two situations of noisy perturbation:

(a) $\delta b_A = 0$ and perturbation only in $N(A^T)$

(b) $||\delta b_A|| = ||\delta b^*_A||$, i.e. total perturbation (in both spaces $R(A)$ and $N(A^t)$)

Assuming that in EM geotomography, $SNR = 20\log(\frac{||b||}{||\delta b||})$ (Signal-to-Noise Ratio) is about 25 dB, we should have $c_{noise} = \frac{||\delta b||}{||b||} \cong 5.5\%$. Taking into account this for all the above situations we have:

| | `Case 1` | `Case 2` |
|---|---|---|
| **(a)** | $\delta b_A = 0$, $\|\|\delta b^*_A\|\| = 2.4874$ | $\delta b_A = 0$, $\|\|\delta b^*_A\|\| = 15.9913$ |
| **(b)** | $\|\|\delta b_A\|\| = \|\|\delta b^*_A\|\| = 1.76$ | $\|\|\delta b_A\|\| = \|\|\delta b^*_A\|\| = 11.3076$ |

All the tested algorithms started from initial guess $x_0 = 0$, thus they gave approximations for the minimal norm solution $x_{LS}$ given by (see also Figures 1 and 2 (right))

$$x_{LS} = P_{R(A^T)}(x_{exact}), \quad \text{where} \quad P_{R(A^T)} = V_{range} V_{range}^T$$

and $V_{range} = [v_1, \ldots, v_r]$ is built from respective left singular vectors of $A$.

### 4.4 Results

The reconstructions have been evaluated with both qualitative and quantitative criteria. We can decide about the quality of the reconstructed images by comparing them with the original ones (Figs. 1, 2 (left)), or with the corresponding ones of the minimal 2-norm (Figs. 1, 2 (right)). The recent comparison makes it possible to detect artifacts which are strongly dependant on the algorithm. The quantitative criteria used here are the same as these described by Censor *et al.* in [4], e.g. the *distance*, *relative error* and *standard deviation* measures.

The *standard deviation* of $x^k$ is computed as follows:

$$\sigma_k \triangleq \frac{1}{\sqrt{N}} ||x^k - \rho_k||, \quad \text{where} \quad \rho_k \triangleq \frac{1}{N} \sum_{j=1}^{N} x_j^k \tag{75}$$

Let $\rho_{exact}$, $\sigma_{exact}$ be the average value and standard deviation computed as above but related to the original image, respectively.

The *distance* between $x^k$ and original image $x_{exact}$ is defined by

$$\delta_k \triangleq \frac{1}{\sigma_{exact}\sqrt{N}} ||x^k - x_{exact}|| \tag{76}$$

Since $\forall j : x_j^{exact} > 0$, the *relative error* of $x^k$ is stated in the following way:

$$\epsilon_k \triangleq \frac{\sum_{j=1}^{N} |x_j^k - x_j^{exact}|}{\sum_{j=1}^{N} x_j^{exact}} \tag{77}$$

The *distance* measure is vulnerable to large discrepancies between the reconstructed and original images and it is nearly insensitive to the small differences, whereas the *relative error* measure treats all differences proportionally. Thus, the reconstructed images with local perturbation of a few pixels will be scored worse with the distance measure than with the relative error one.

In our experiments, the **KR** and **CEG** algorithms are run with different relaxation parameters, whereas the results concerning the **KERP** algorithm are presented only for its optimal parameters. They are optimal in the sense of minimal distance (76) within a fixed range of iterations. The optimal parameters were evaluated on the basis of multiple runs with consecutive values of sampled $\alpha$ and $\omega$, separately for each case (1 and 2). The question of an a priori (theoretical) estimation of the relaxation parameters is still open.

All the reconstructed images are presented only for the following fixed iterative steps: 20, 50, 150, 250, 500 and 1000. Figs. 3-8 present the images reconstructed with the **KERP**, **KR** and **CEG** algorithms (with $\lambda_0$ being the right bound of the interval from (48)) applied to `Case 1`. The images related to `Case 2` are shown in Figs. 12-17. Figs. 9-11 illustrate the plots of the *distance*, *relative error* and *standard deviation* measures for `Case 1`, respectively. The similar plots related with `Case 2` are presented in corresponding Figs. 18-20.

## 5 Conclusions

The results in Section 4.4 prove the theoretical considerations discussed in Section 4.1. Figs. 9, 10 for `Case 1` and Figs. 18, 19 for `Case 2` show that for **KERP** applied to noisy data with $\delta b_A = 0$, $z^*(x^0 = 0)$ in (71) converges to $x_{LS}$, and hence the distance in (72) approaches to zero. Increasing a number of iterations in **KERP** for $\delta b_A = 0$, we observe a gradual improvement in the quality of the reconstructed images (see Fig. 3 and Fig. 12), whereas this does not hold for **KR** and **CEG** (see Figs. 5, 7, 14, 16). This is in agreement with our expectations because for **KR** the distance in

(70) depends on both perturbation $\delta b_A$ and $\delta b_A^*$ and in our case we cannot approach to $x_{LS}$ closer than in distance $||G\delta b_A^*||$. The similar conclusion can be drawn for the **CEG** algorithm, considering (55).

For $||\delta b_A|| = ||\delta b_A^*||$, the best images obtained with **KERP** appear at $k = 250$ (see Fig. 4 and Fig. 13). Comparing them with the ones reconstructed with **KR** and **CEG** even for small parameters ($\omega$, $\lambda$), we can notice that **KERP** scores significantly - especially for higher iterations. This is also due to the fact that distance in (72) does not depend on $\delta b_A^*$. For $k > 250$, the images reconstructed with **KERP** deteriorate so slightly that we can easily control it, even for large $||\delta b_A||$. For a smaller number of iterations, the **KR** and **CEG** algorithms have the *relative error* measures of lower values (see Figs. 10 and 19) than **KERP**, whereas according to the *distance* measure, **KERP** is still better (see Figs. 9 and 18). This is because **KERP** gives smoother images and noise affected images with **KR** and **CEG** are poorly evaluated by the *distance* measure. For initial iterates, **KERP** behaves like the other algorithms with possibly large relaxation parameters, e.g. it appears to be fast-convergent.

Summing up, a superiority of **KERP** over the other tested methods is significant. For inconsistent inverse problems appearing in EM geotomography, **KERP** gives the approximations closer to the minimal norm solution than the other methods. This is due to the fact that the limit point in **KERP** is not affected by perturbation lying in the orthogonal complement of the range of the problem's matrix. This feature is highly desirable here because in EM geotomography, additive perturbation have a Gaussian distribution whereby we have as many undesirable components in the range as in its orthogonal complement.

## Acknowledgments

The paper was supported by the NATO Grant PST.CLG.977924 and the DAAD grant that one of the authors had as a visiting professor at the Friedrich-Alexander-University Erlangen-Nuremberg, Germany, in the period October 2002 to August 2003.

# References

[1] Björck A., *Numerical methods for least squares problems*, SIAM Philadelphia, 1996.

[2] Censor Y., *Finite series-expansion reconstruction methods*, Proc. IEEE, **71** (1983), 409–419.

[3] Censor Y., Eggermont P.B., Gordon D., *Strong Underrelaxation in Kaczmarz's Method for Inconsistent Systems*, Numer. Math., **41**(1983), 83-92.

[4] Censor Y., Gordon D., Gordon R., *Component averaging: An efficient iterative parallel algorithm for large and sparse unstructured problems*, Parallel Computing, **27** (2001), 777-808.

[5] Censor Y., Stavros A. Z., *Parallel optimization: theory, algorithms and applications*, "Numer. Math. and Sci. Comp." Series, Oxford Univ. Press, New York, 1997.

[6] Dax, A., *An open question on cyclic relaxation*, to appear in *B I T*.

[7] Dines K. A., Lytle R. J., *Computerized geophysical tomography*, Proc. IEEE, **67** (1979), 1065–1073.

[8] Golub G. H., van Loan C. F., *Matrix computations*, The John's Hopkins Univ. Press, Baltimore,1983.

[9] Gulliksson, M. et al., *Perturbation bounds for constrained and weighted least squares problems*, Linear Alg. Appl., **349**(2002), 221-232.

[10] Hanke M., Niethammer, W., *On the acceleration of Kaczmarz's method for inconsistent linear systems*, Linear Alg. Appl., **130**(1990), 83-98.

[11] Herman G.T. , *Image Reconstruction from Projections: The Fundamentals of Computerized Tomography*, Academic Press Inc., New York, 1980.

[12] Natterer F., *The Mathematics of Computerized Tomography*, John Wiley and Sons, New York, 1986.

[13] Popa C., *Least-squares solution of overdetermined inconsistent linear systems using Kaczmarz's relaxation*; Intern. J. Comp. Math., **55**(1995), 79-89.

[14] Popa C., *Characterization of the solutions set of inconsistent least-squares problems by an extended Kaczmarz algorithm*; Korean Journal on Comp. and Appl. Math., **6(1)**(1999), 51-64.

[15] Popa C., *Extensions of block-projections methods with relaxation parameters to inconsistent and rank-deficient least-squares problems*;
*B I T*, **38(1)**(1998), 151-176.

[16] Popa C., *Block-projections algorithms with blocks containing mutually orthogonal rows and columns*, *B I T* , **39(2)**(1999), 323-338.

[17] Tanabe K., *Projection Method for Solving a Singular System of Linear Equations and its Applications*, Numer. Math., **17**(1971), 203-214.

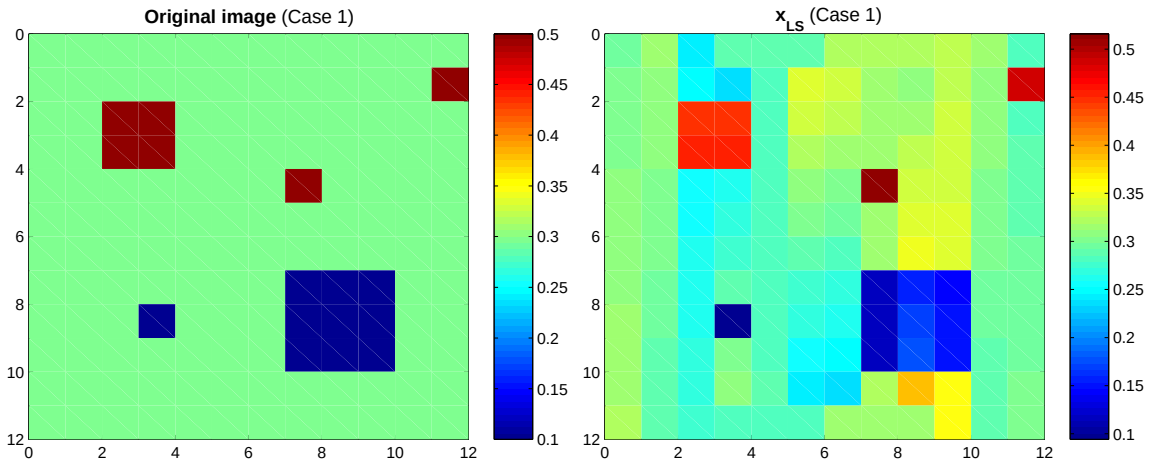


Figure 1: Original image - Case 1 (left), $P_{R(A^T)}(x_{exact})$ (right)

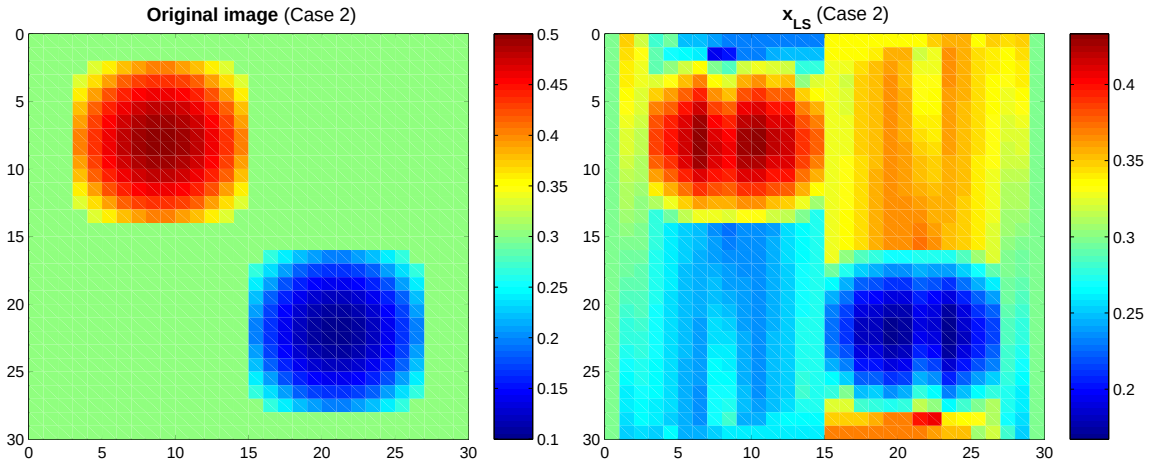


Figure 2: Original image - Case 2 (left), $P_{R(A^T)}(x_{exact})$ (right)

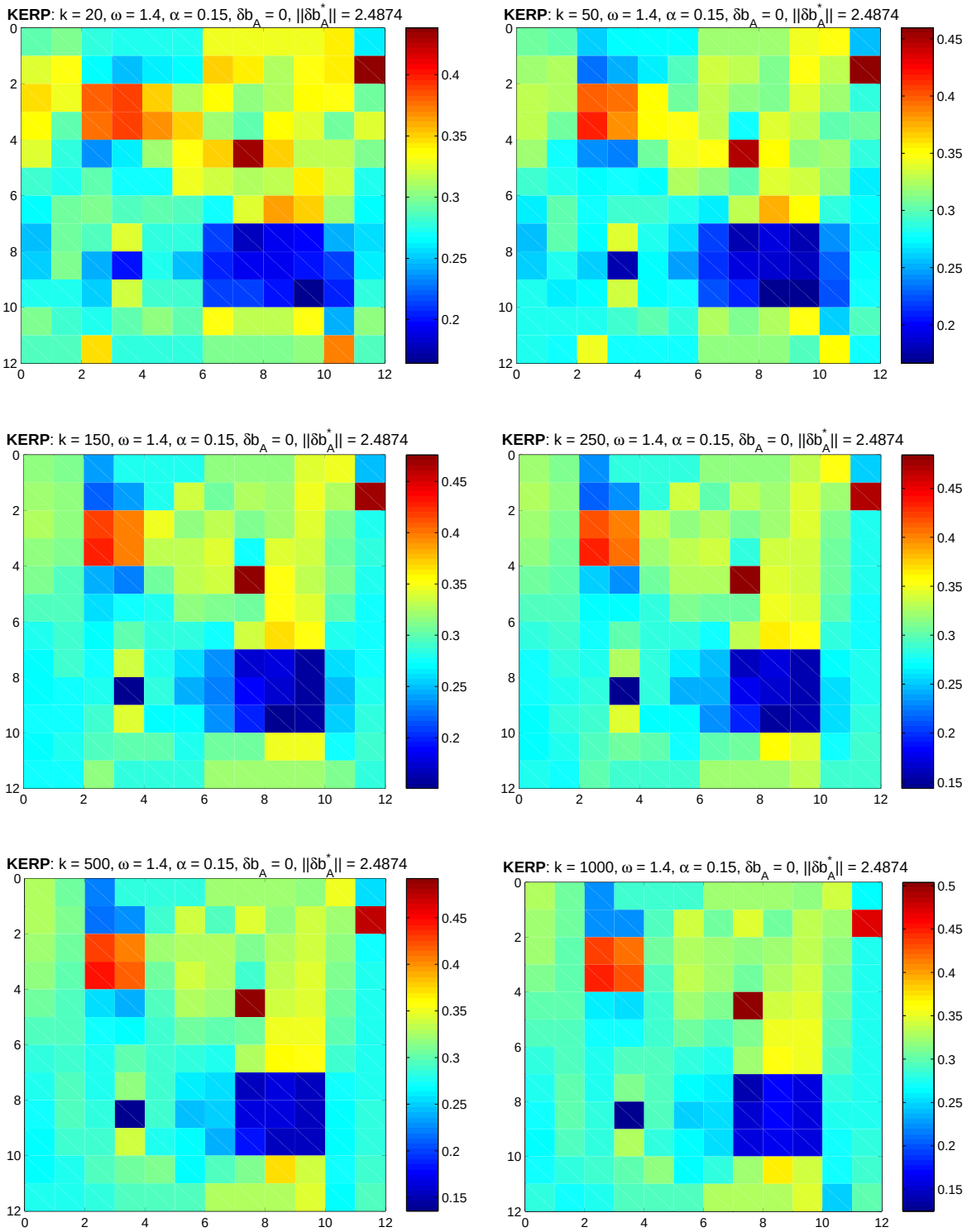


Figure 3: Image reconstruction with **KERP**; Case 1; $\delta b_A = 0$

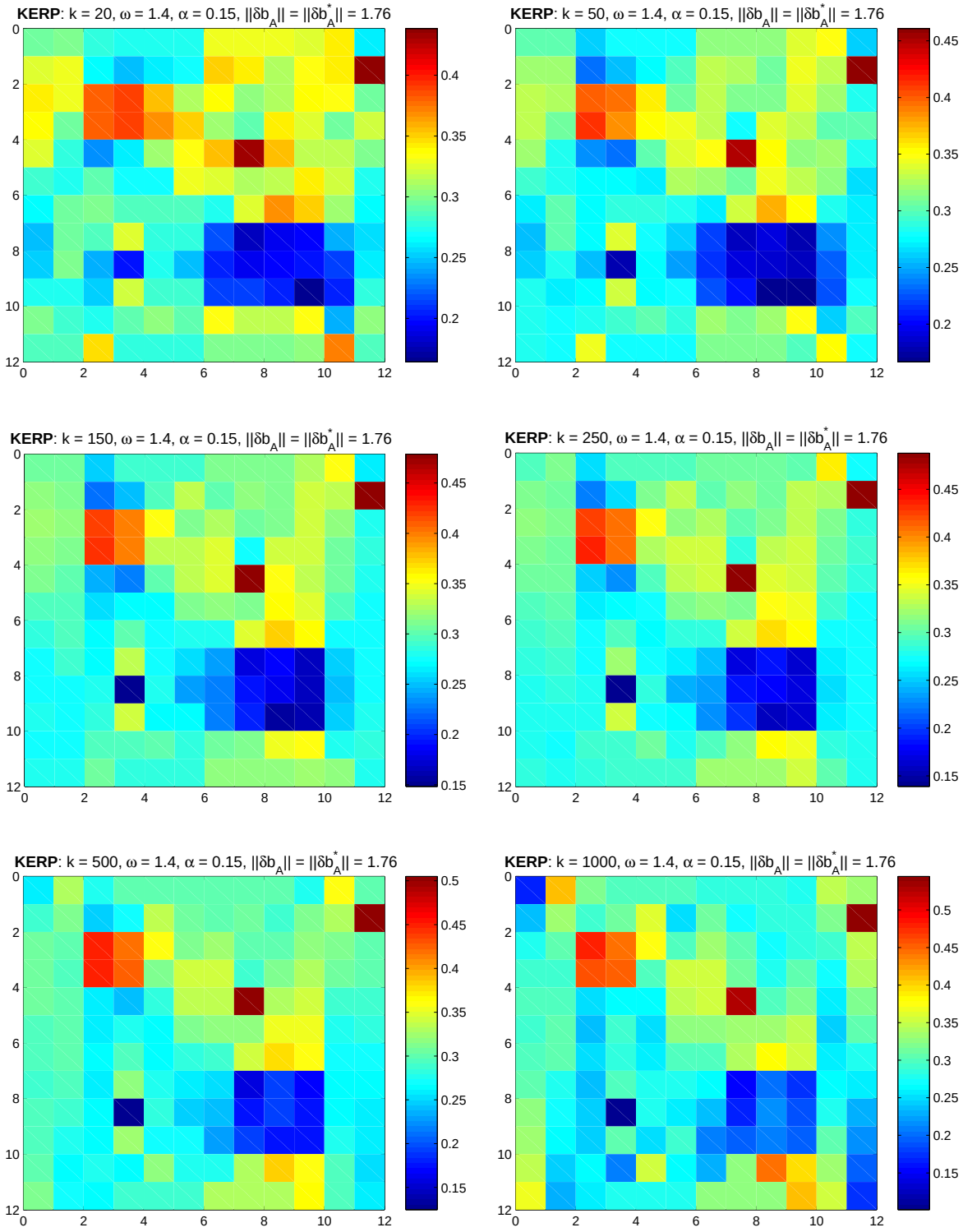


Figure 4: Image reconstruction with **KERP**; Case 1; $\| \delta b_A \| = \| \delta b_A^* \|$

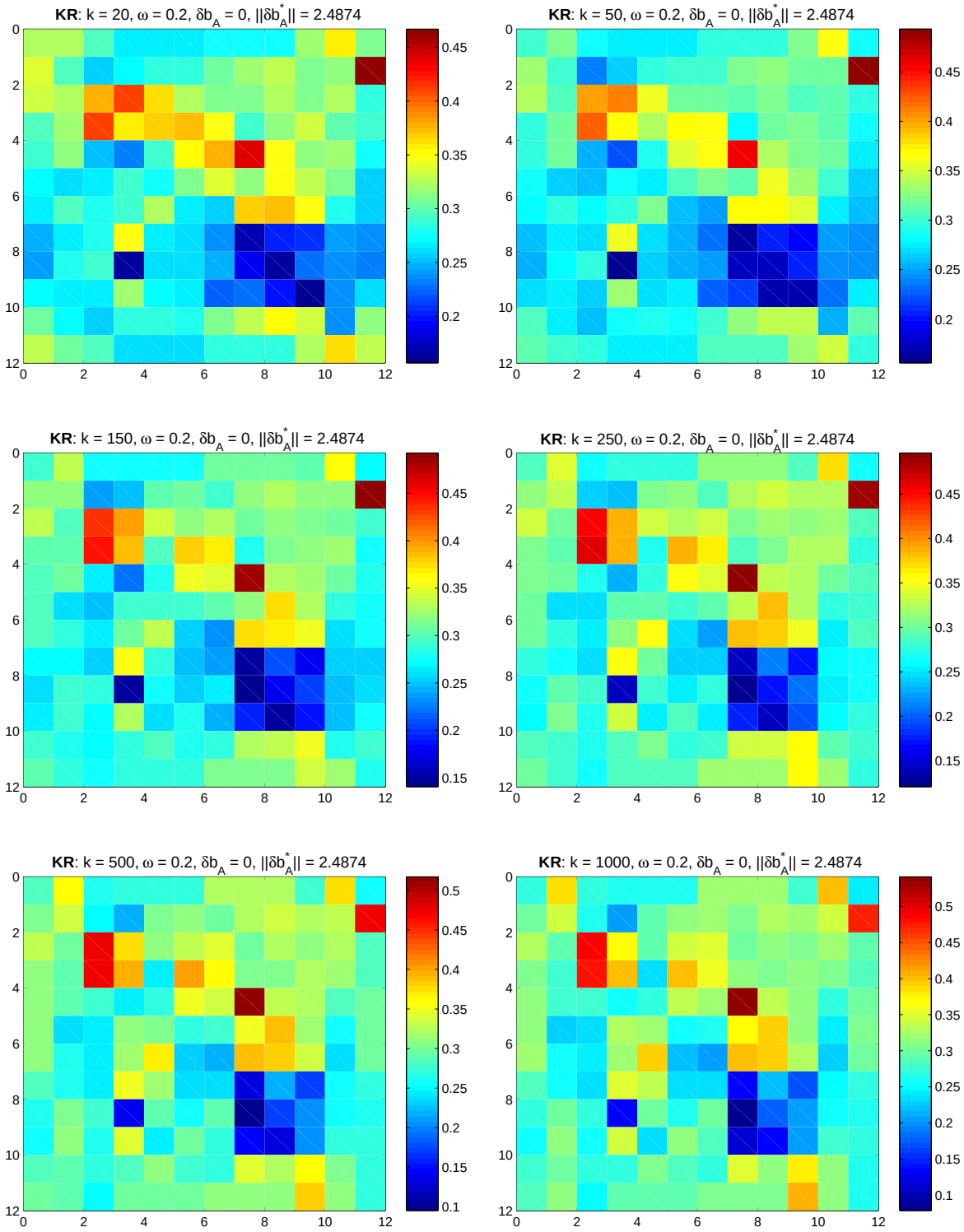


Figure 5: Image reconstruction with **KR**; Case 1; $\delta b_A = 0$

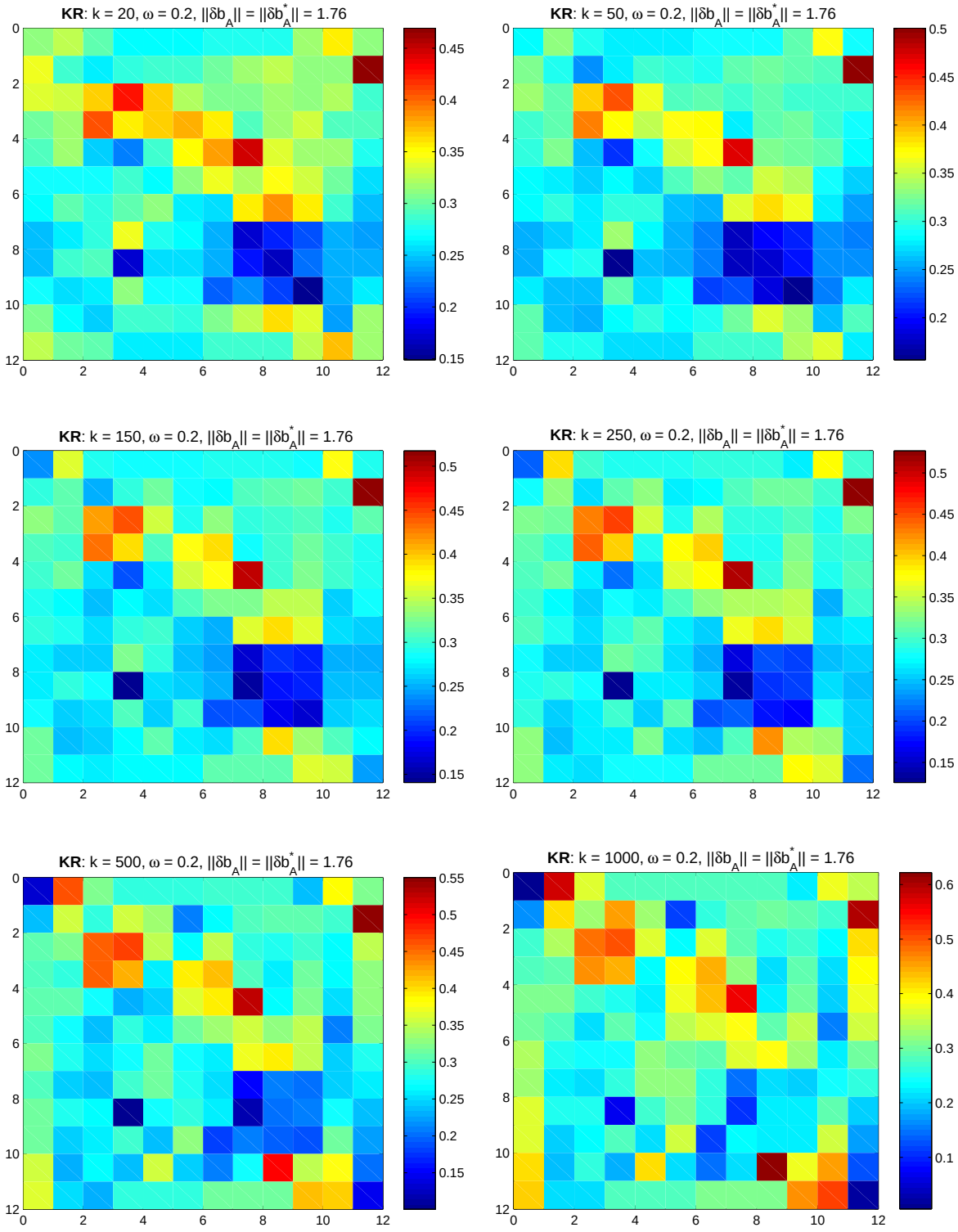


Figure 6: Image reconstruction with **KR**; Case 1; $\| \delta b_A \| = \| \delta b_A^* \|$

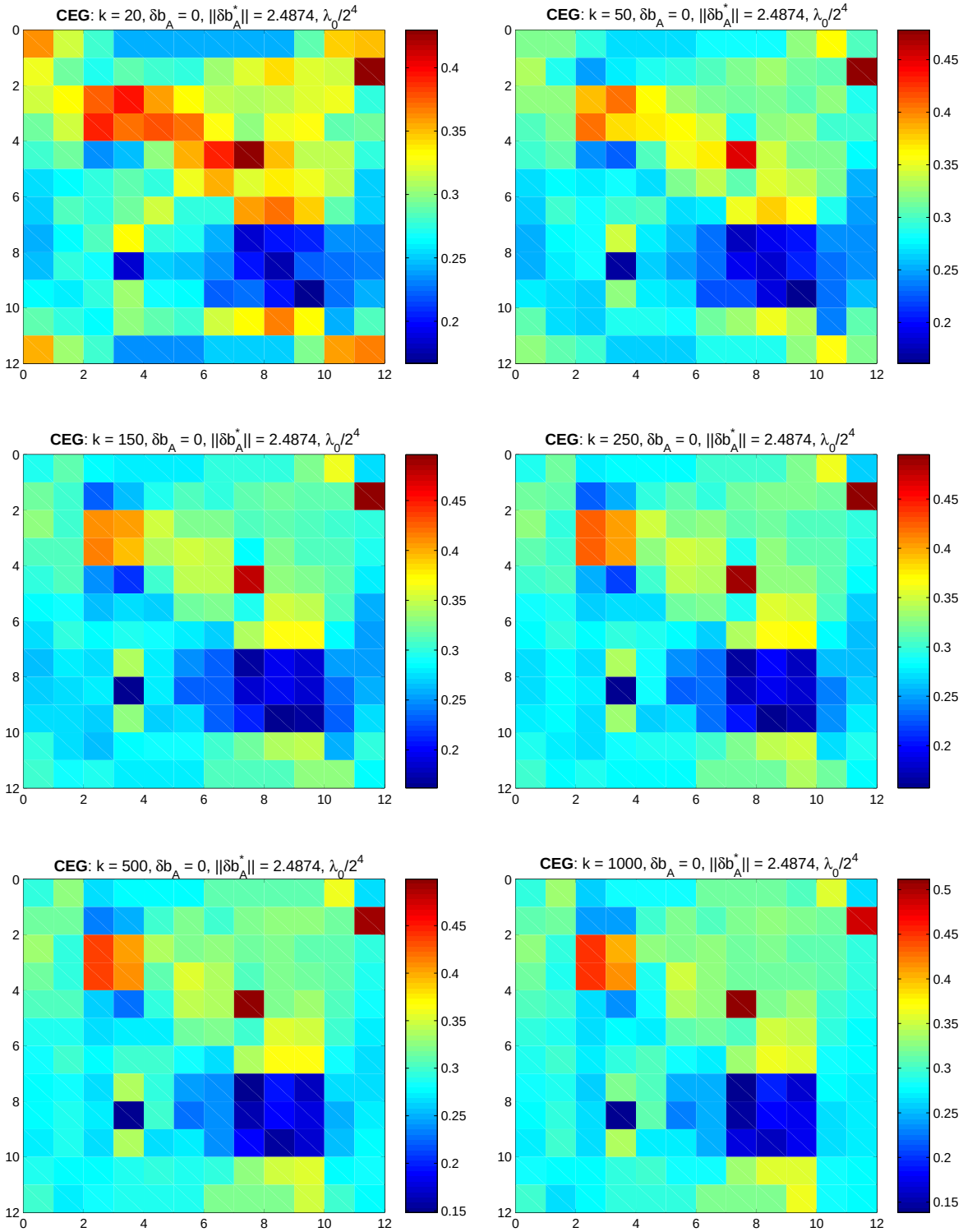


Figure 7: Image reconstruction with **CEG**; Case 1; $\delta b_A = 0$

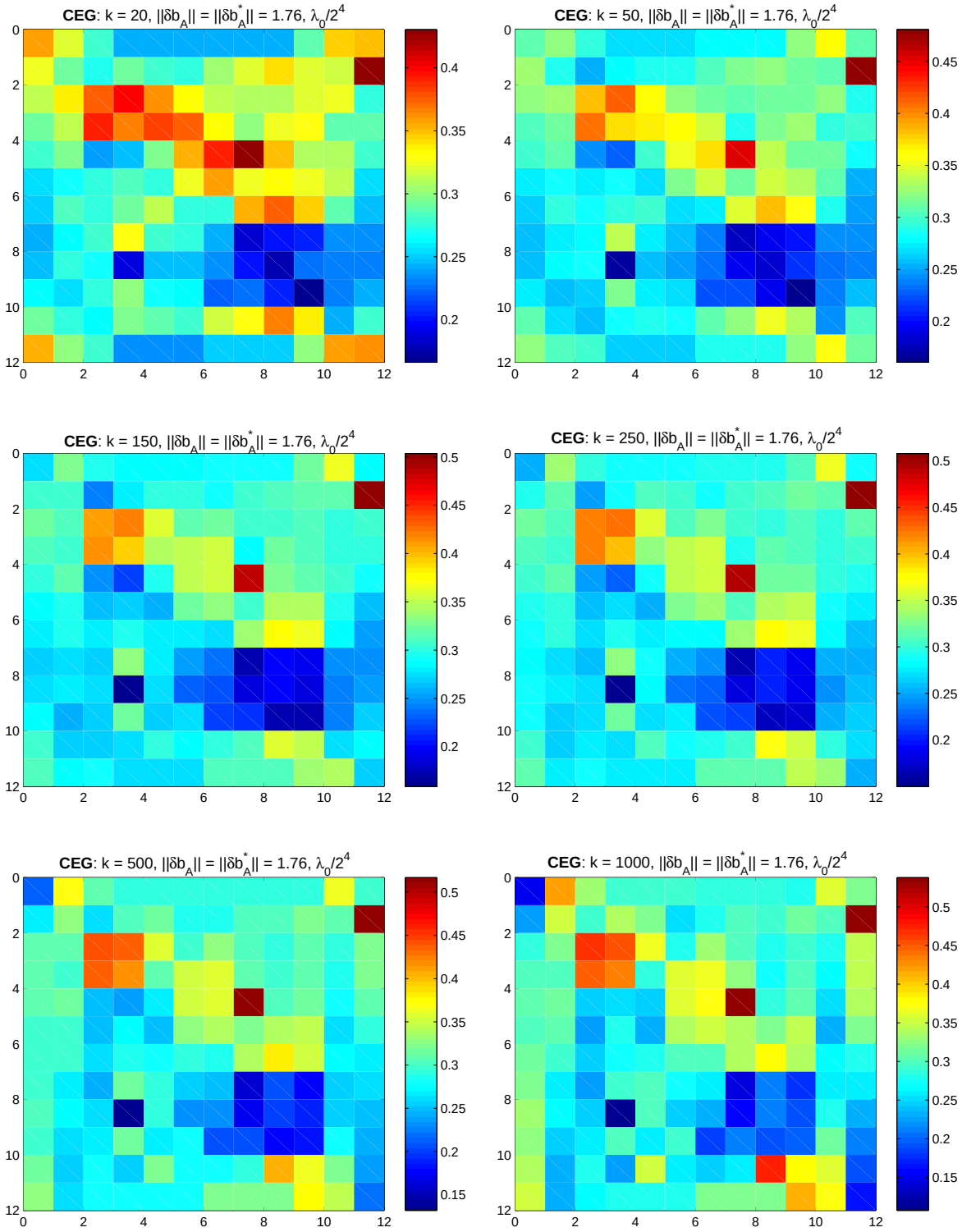


Figure 8: Image reconstruction with **CEG**; Case 1; $\| \delta b_A \| = \| \delta b_A^* \|$

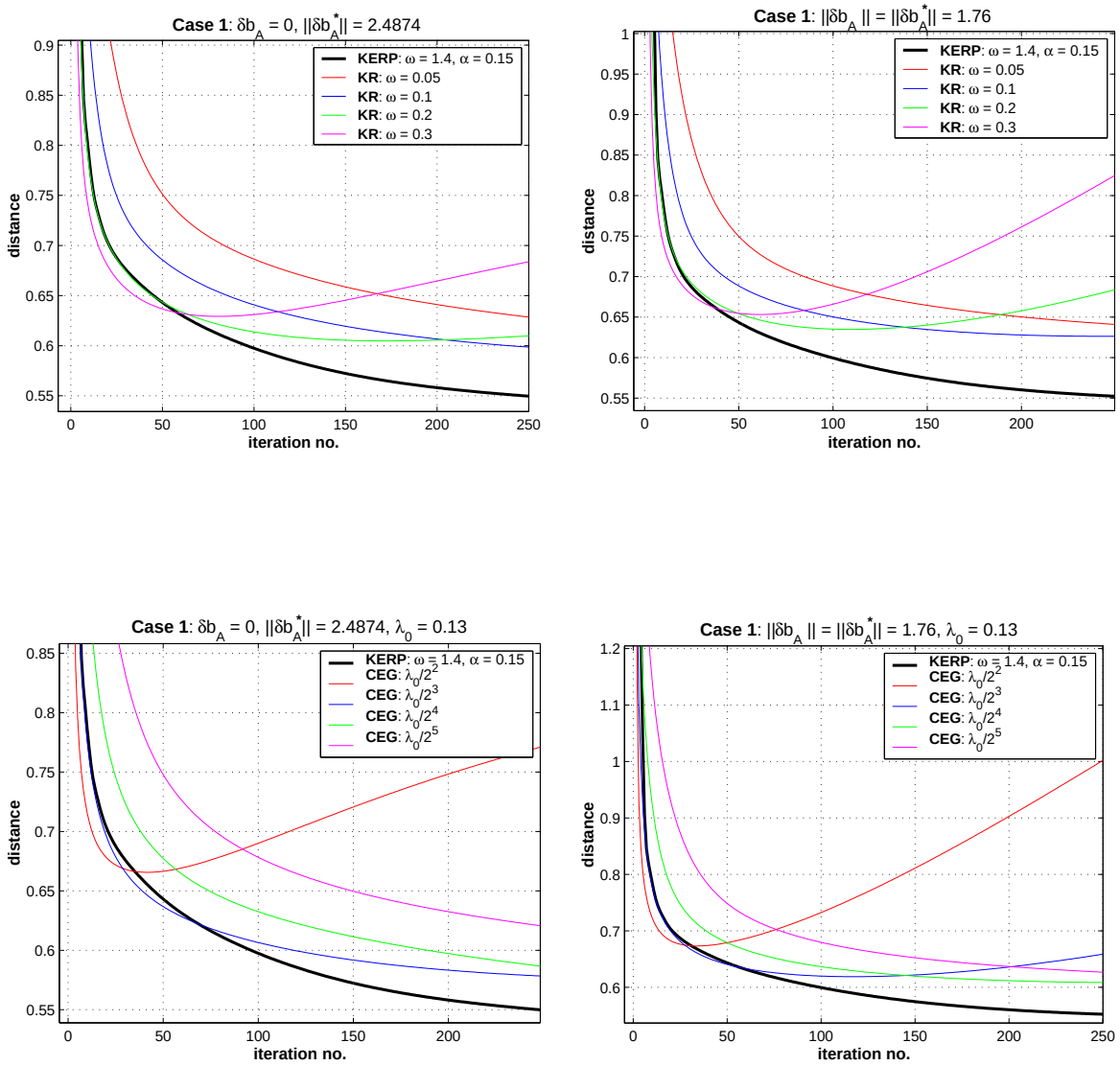


Figure 9: Distance between $x^k$ and original image $x_{exact}$ for Case 1

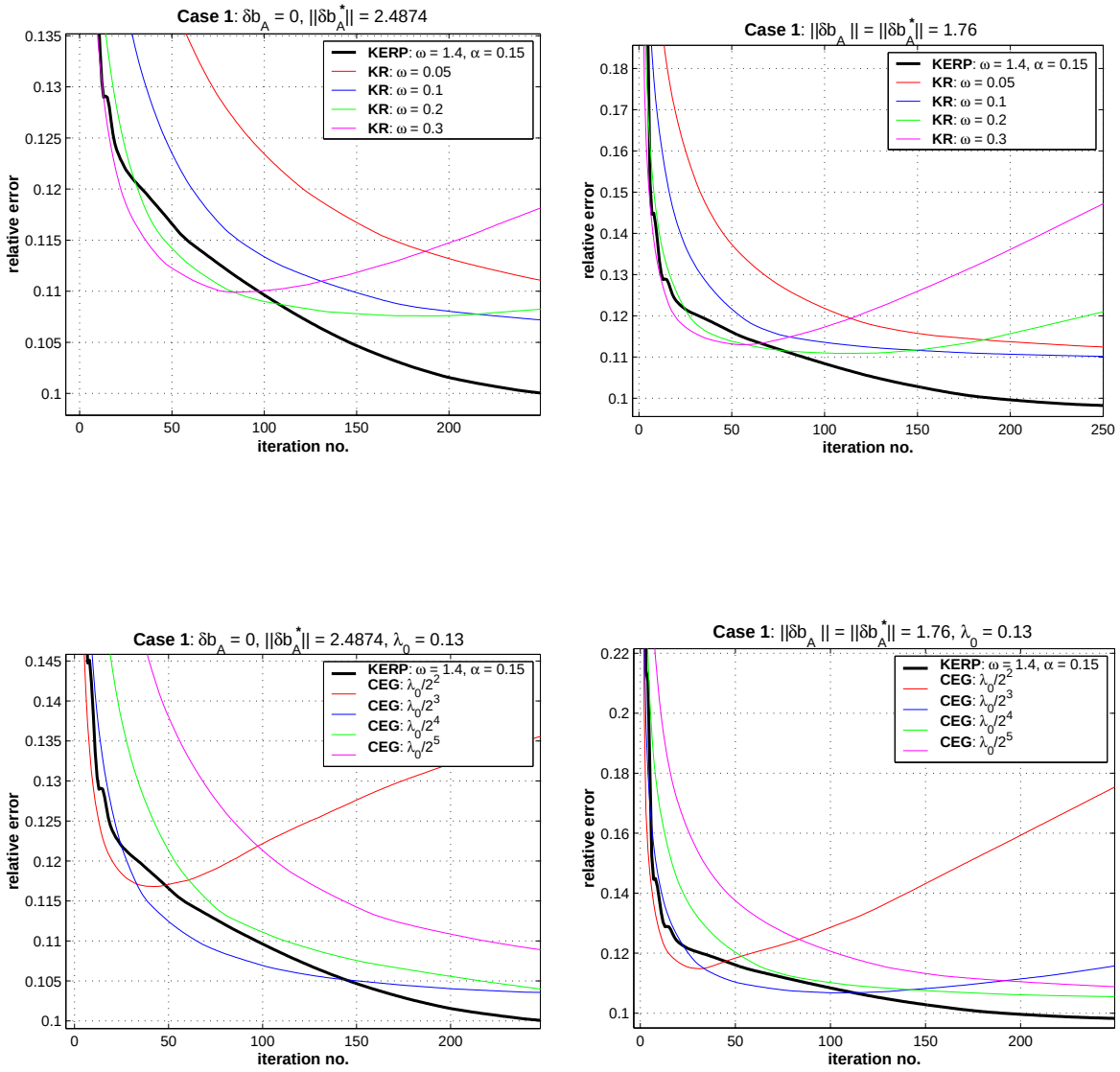


Figure 10: Relative error between $x^k$ and original image $x_{exact}$ for Case 1

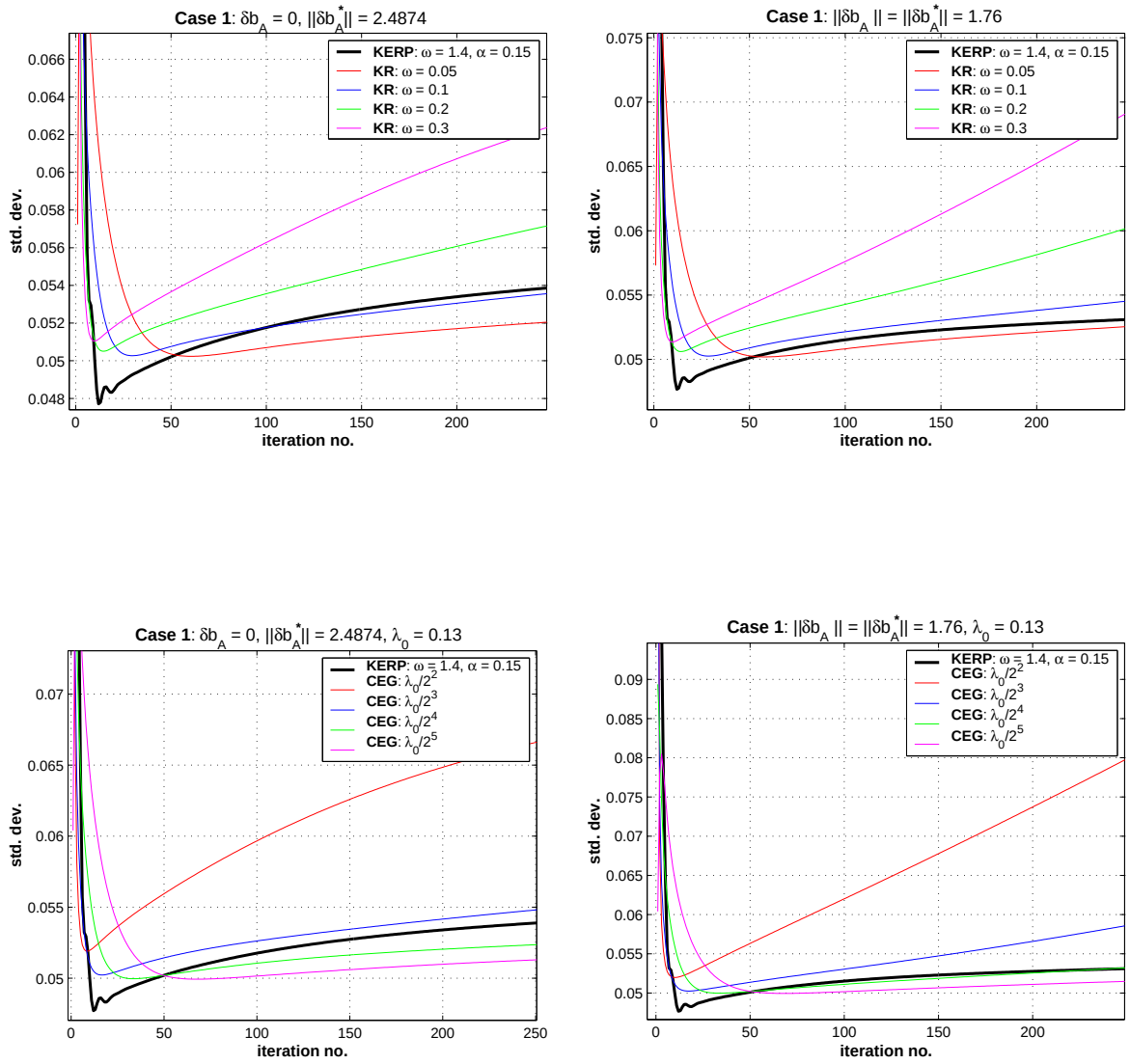


Figure 11: Standard deviation of reconstructed image $x^k$ for Case 1

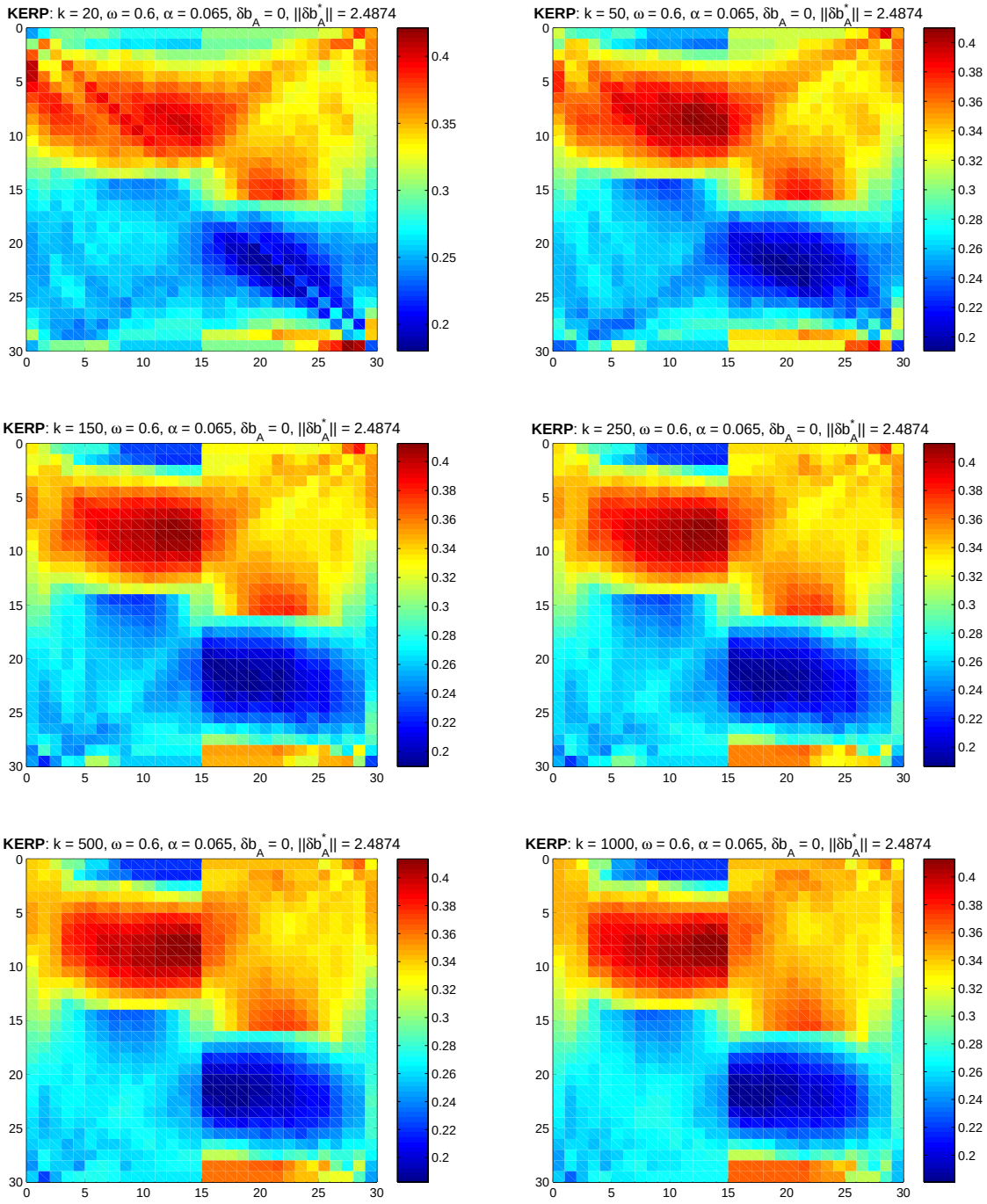


Figure 12: Image reconstruction with **KERP**; Case 2; $\delta b_A = 0$

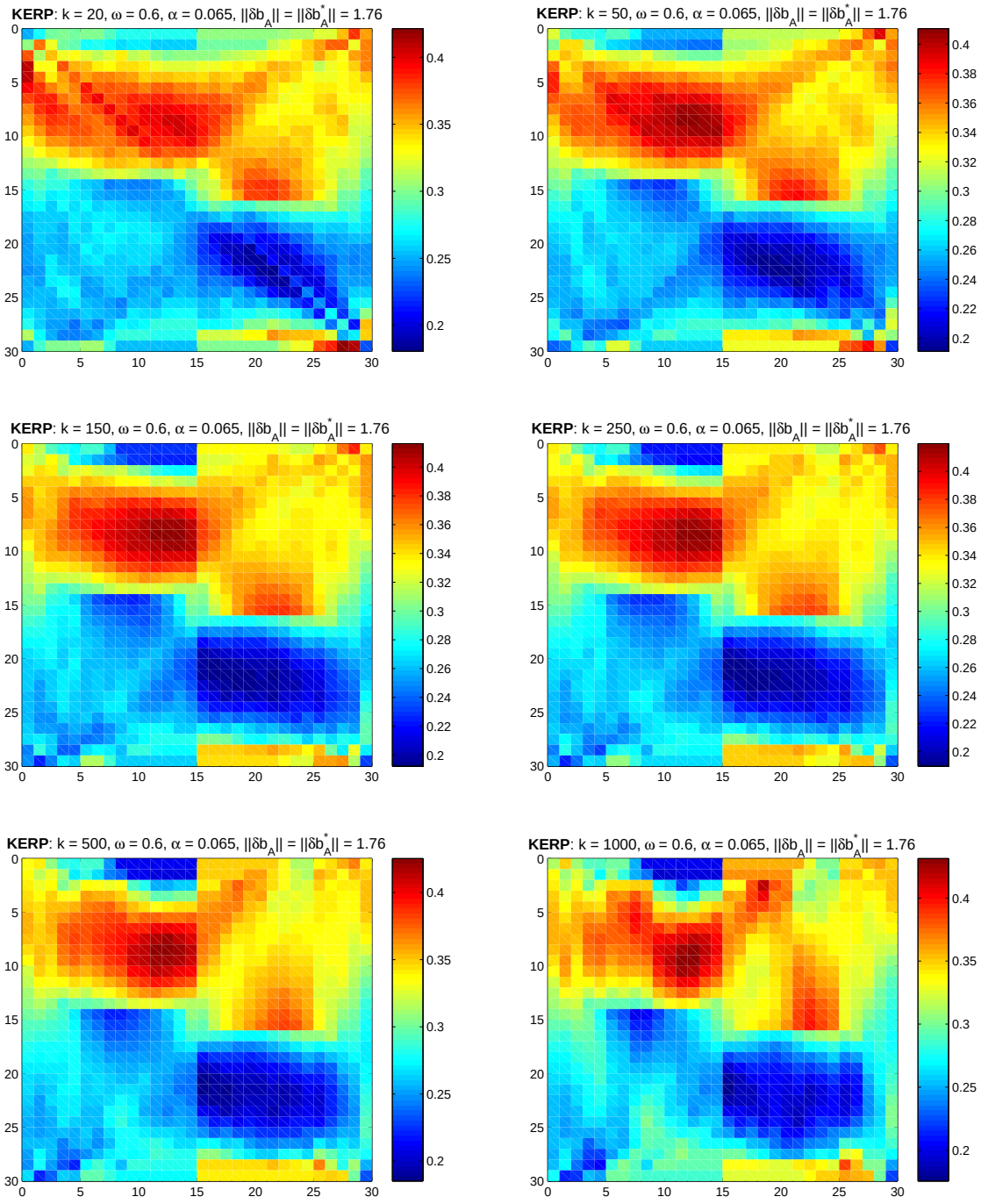


Figure 13: Image reconstruction with **KERP**; Case 2; $\| \delta b_A \| = \| \delta b_A^* \|$

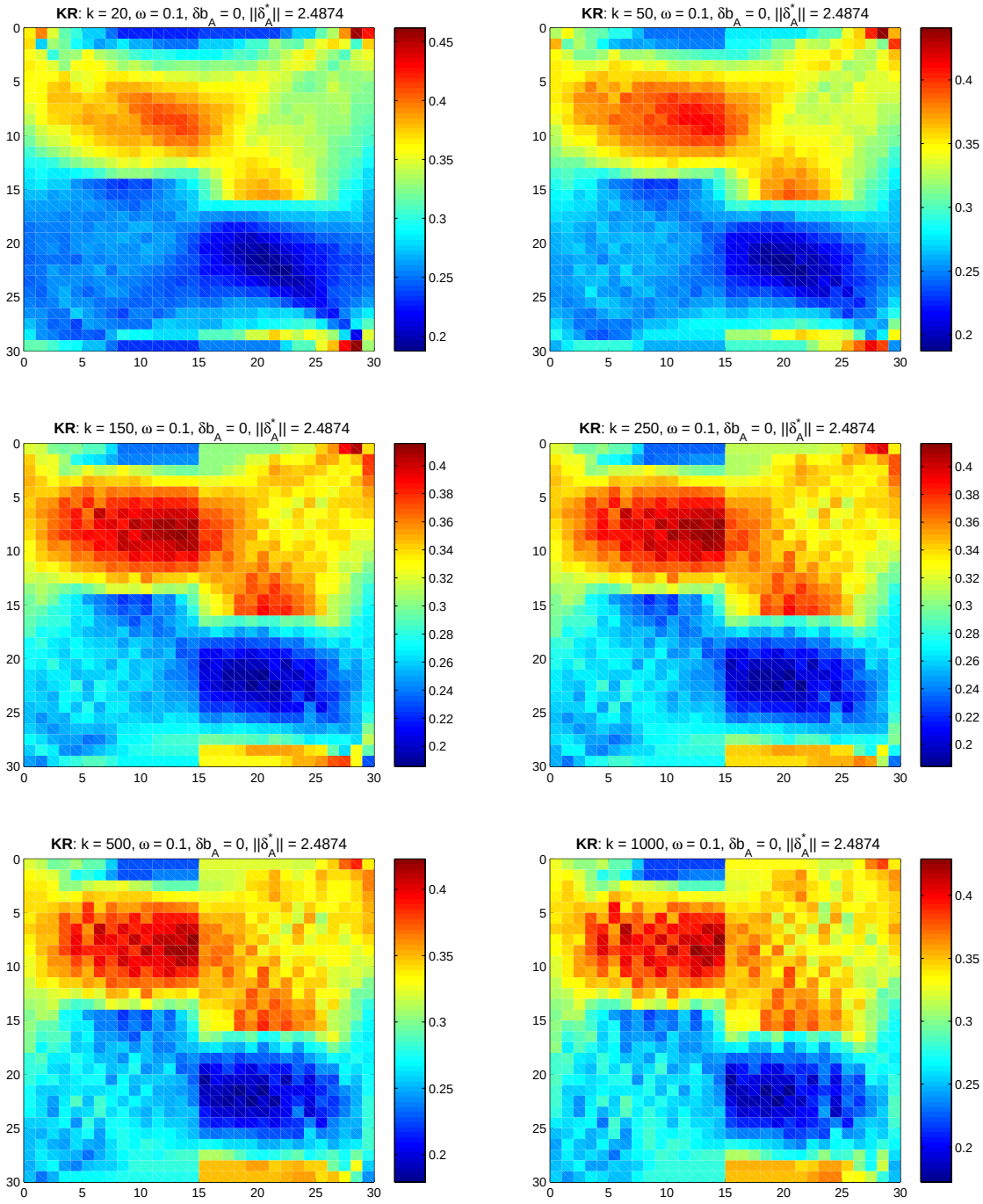


Figure 14: Image reconstruction with **KR**; Case 2; $\delta b_A = 0$

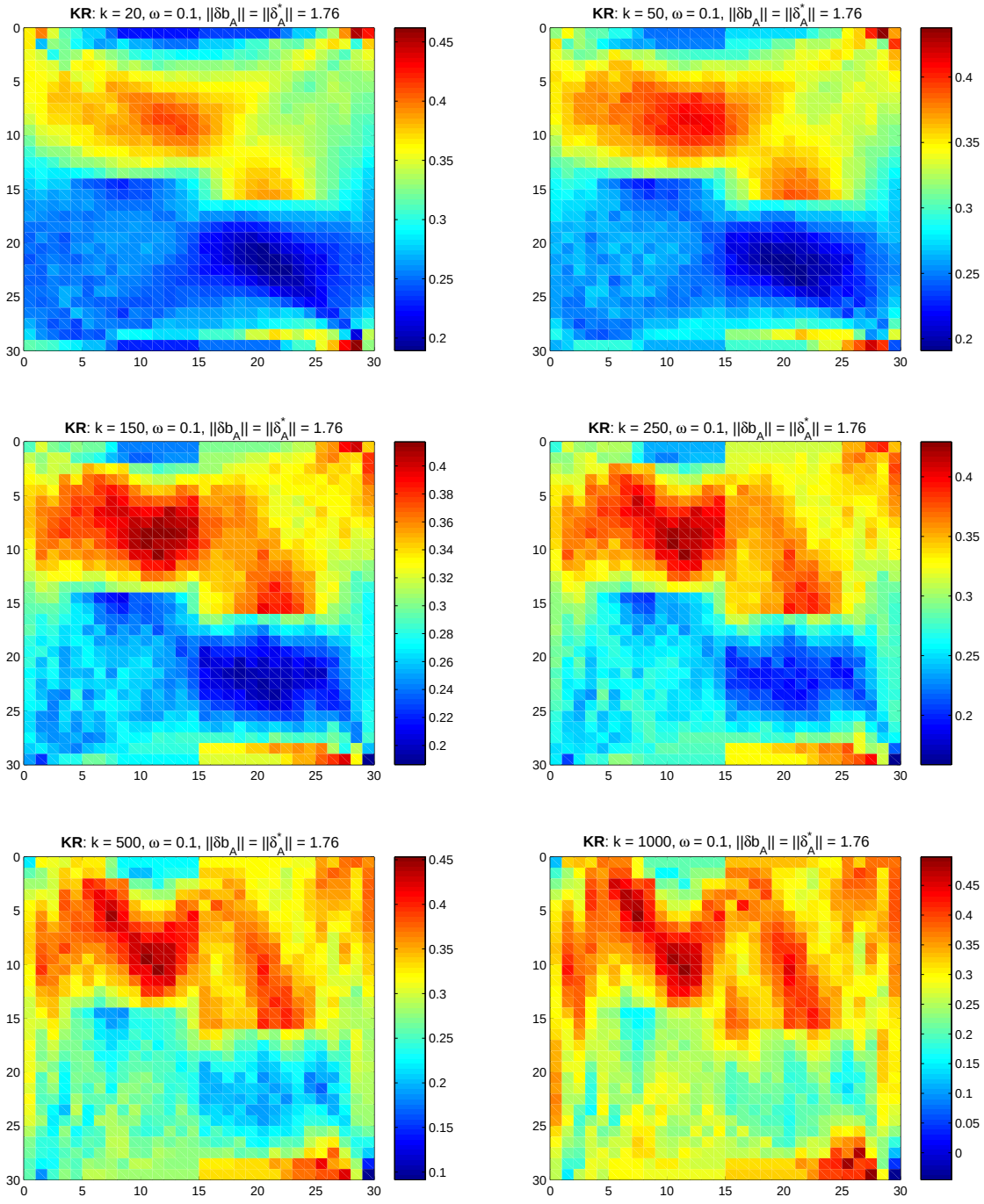


Figure 15: Image reconstruction with **KR**; Case 2; $\| \delta b_A \| = \| \delta b_A^* \|$

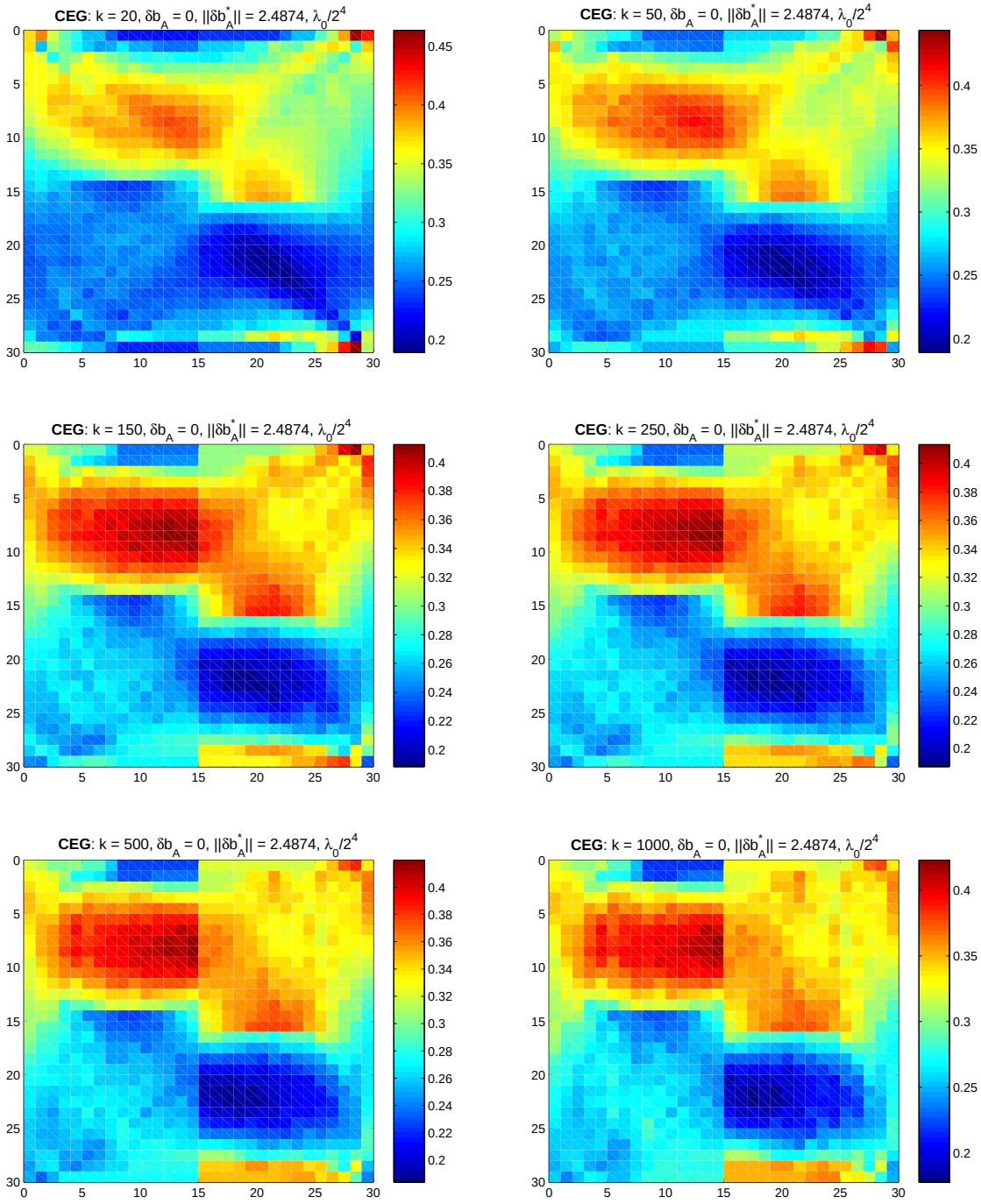


Figure 16: Image reconstruction with **CEG**; Case 2; $\delta b_A = 0$

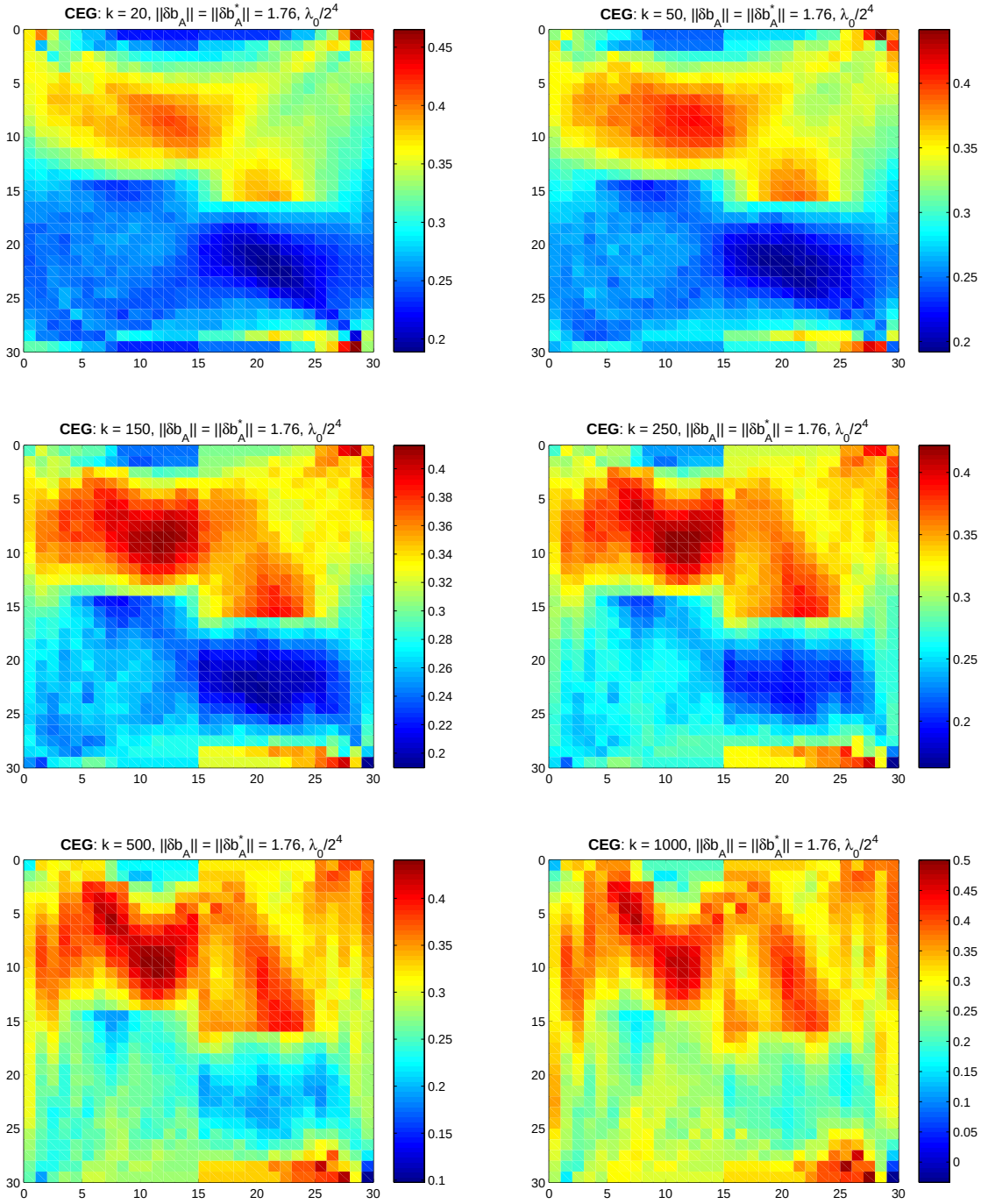


Figure 17: Image reconstruction with **CEG**; Case 2; $\| \delta b_A \| = \| \delta b_A^* \|$

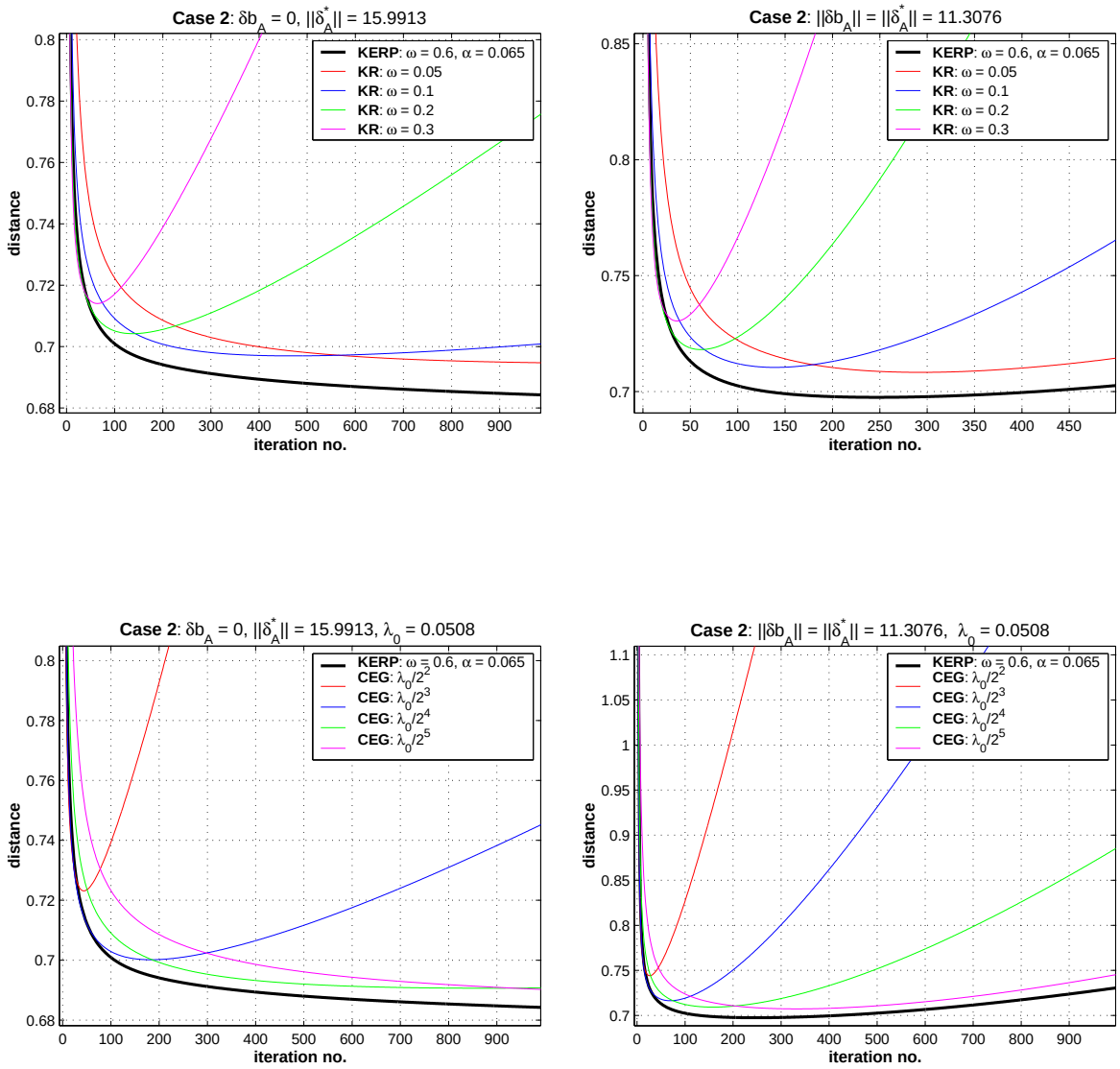


Figure 18: Distance between $x^k$ and original image $x_{exact}$ for Case 2

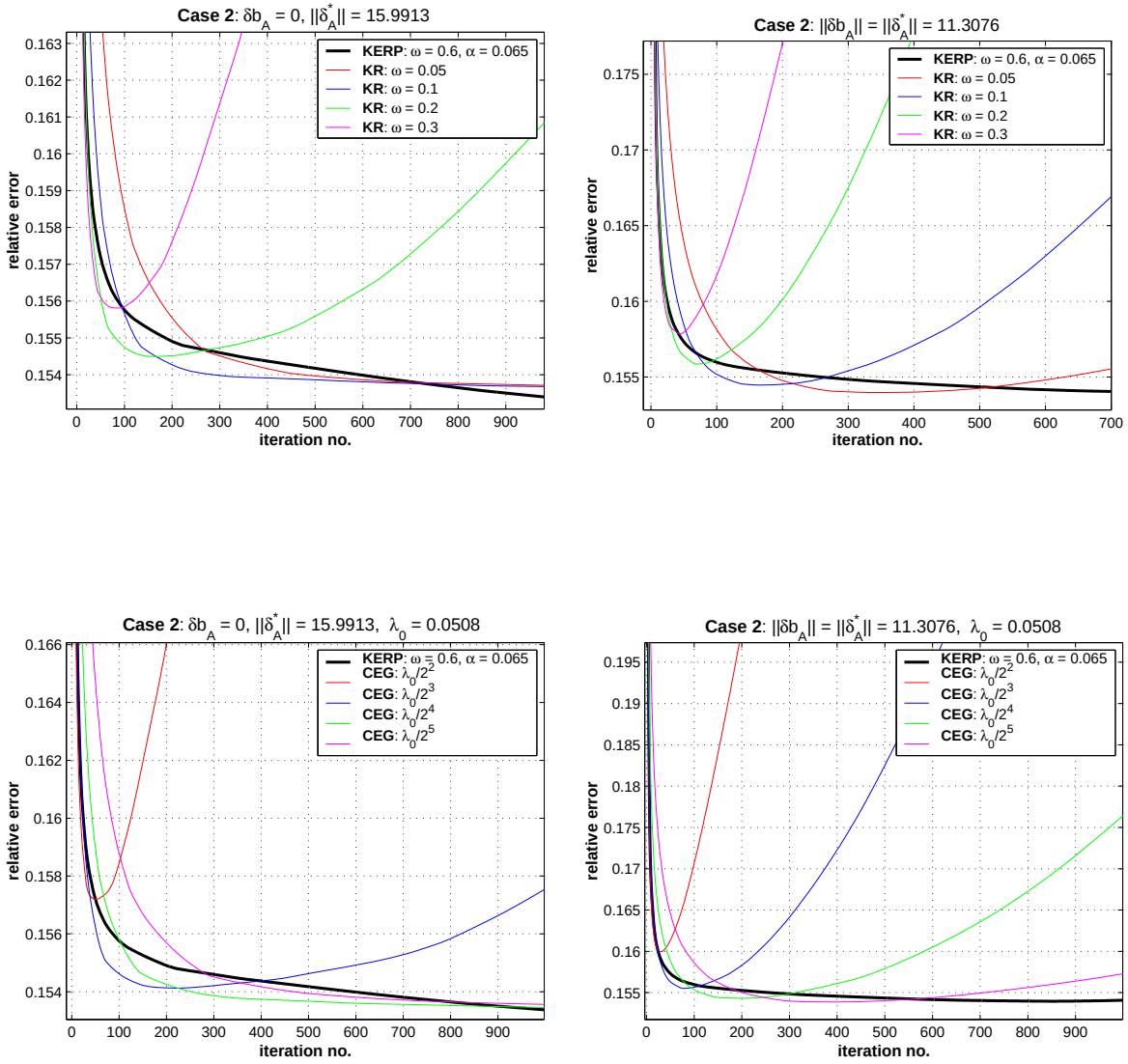


Figure 19: Relative error between $x^k$ and original image $x_{exact}$ for Case 2

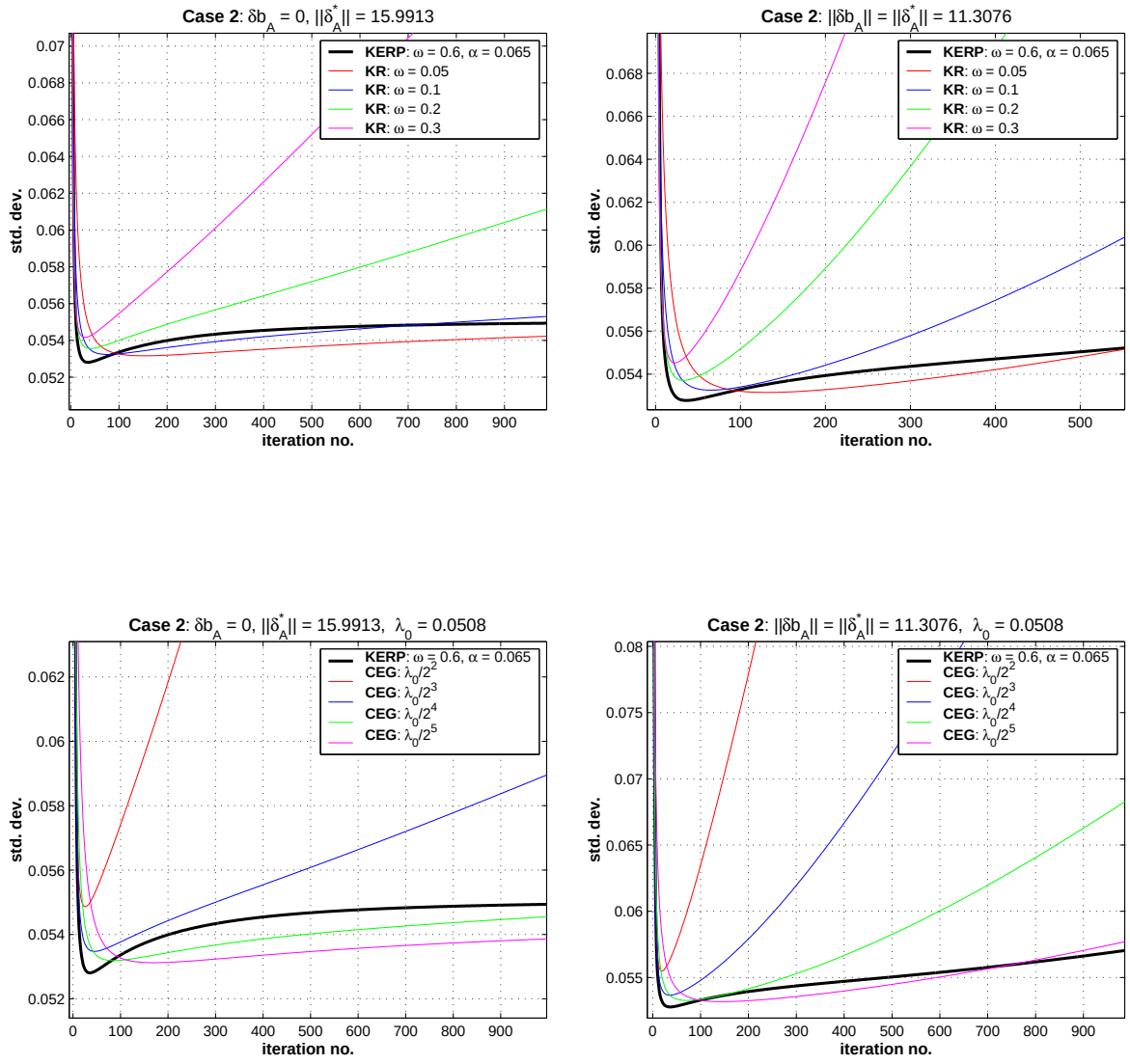


Figure 20: Standard deviation of reconstructed image $x^k$ for Case 2