

**Тема дипломної роботи:
«Відновлення зображень із
застосуванням паралельних технологій
середовища MatLab»**

Автор: Синицький Олекса Ігорович

Науковий керівник: к.т.н., доц. Карпенко Євген Юрійович

Задача відновлення кольорових зображень засобами середовища Matlab

- ▶ удосконалення використання алгоритму SVD для відновлення пошкоджених зображень, інтеграція алгоритму обертань Якобі;
- ▶ реалізація удосконаленого алгоритму;
- ▶ розробка інтерфейсу перегляду результатів роботи програми.

Огляд існуючих рішень поставленої задачі

- ▶ Фільтр Вінера
- ▶ Ранжуючі фільтри
- ▶ Медіанні фільтри
- ▶ Рекурентні фільтри
- ▶ Інверсна фільтрація

Спільне:

Видаляють певні типи шуму.

Недоліки:

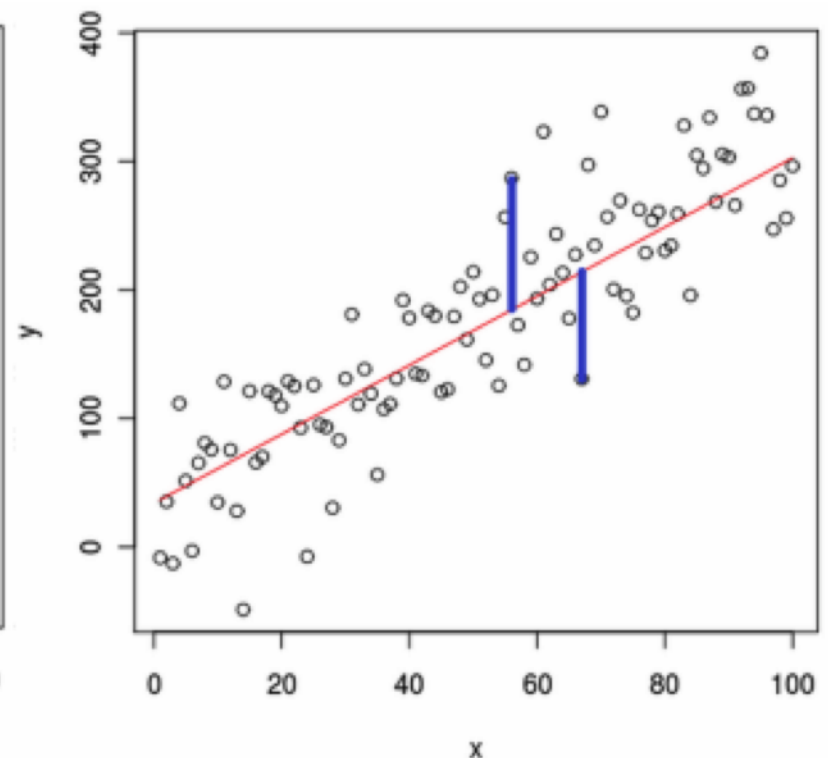
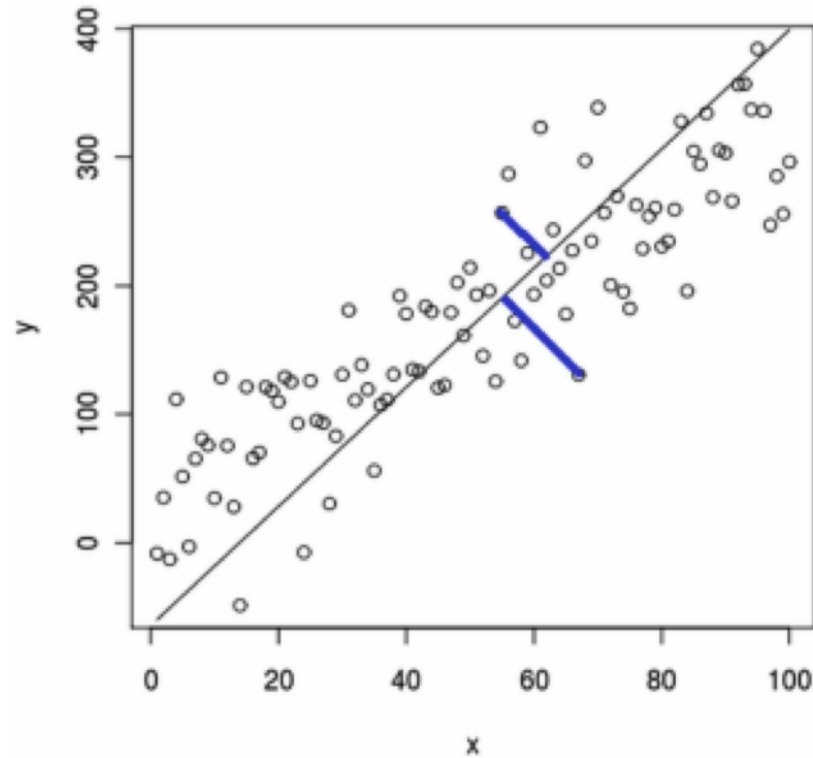
Не охоплюють усі види відновлення, конкретизовані лише під певні типи пошкоджень.



Сингулярні вектори

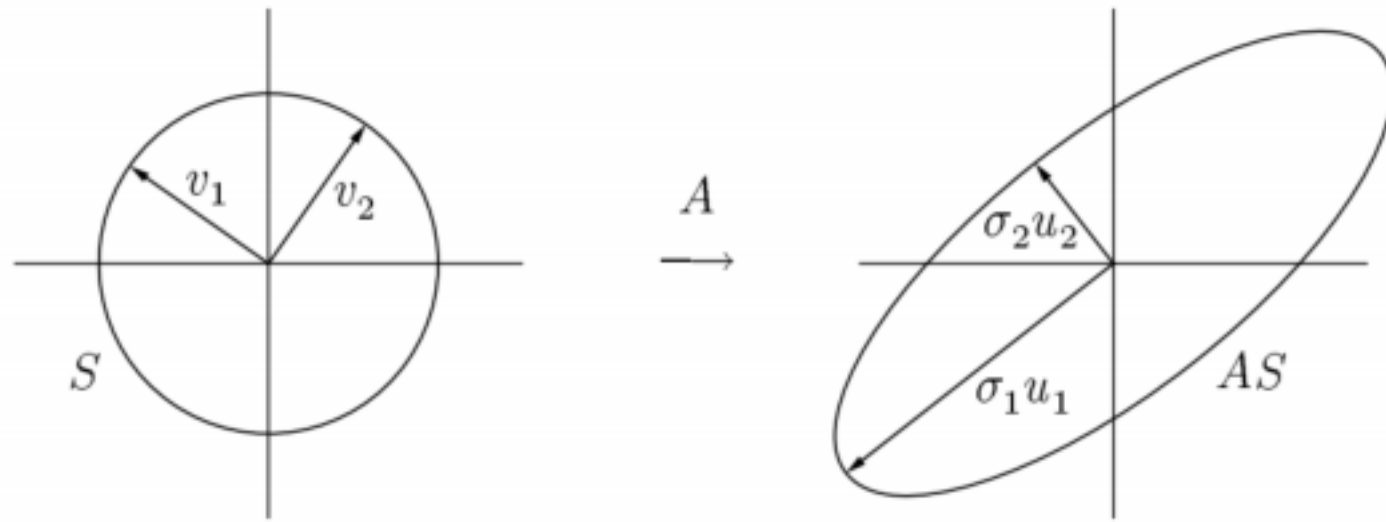
$$A_{n \times d} = \begin{pmatrix} a_1^{(1)} & a_2^{(1)} & \cdots & a_d^{(1)} \\ a_1^{(2)} & a_2^{(2)} & \cdots & a_d^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ a_1^{(n)} & a_2^{(n)} & \cdots & a_d^{(n)} \end{pmatrix}$$

Матричне представлення



Мінімізовані похибки

Геометричне значення сингулярного розкладу



$$A = U\Sigma V^T,$$

U, V – ортогональні матриці Σ – діагональна матриця

Результат сингулярного разложения

$$A = U \begin{matrix} \diagdown & & \\ & E & \\ & & \diagup \\ 0 & & & 0 \end{matrix} V^T$$

The diagram illustrates the Singular Value Decomposition (SVD) of a matrix A . It shows the equation $A = U \Sigma V^T$. Matrix A is represented by a vertical rectangle. Matrix U is also a vertical rectangle. Matrix Σ is a square matrix with a diagonal line from the top-left to the bottom-right, containing the letter E in the center. There are zeros in the top-right and bottom-left corners. Matrix V is a square rectangle, and its superscript T is located to the upper right of the matrix.

Алгоритм обертань Якобі

$$B = \bar{U}\bar{\Sigma}\bar{V}^T$$

$$M^T M = \begin{pmatrix} \alpha & \gamma \\ \gamma & \beta \end{pmatrix}$$

$$\begin{bmatrix} * & & \dots & & * \\ & \ddots & & & \\ & & a_{kk} & \dots & a_{kl} \\ \vdots & & \vdots & \ddots & \vdots \\ & & a_{lk} & \dots & a_{ll} \\ & & & \dots & * \end{bmatrix} \rightarrow \begin{bmatrix} * & & \dots & & * \\ & \ddots & & & \\ & & a'_{kk} & \dots & 0 \\ \vdots & & \vdots & \ddots & \vdots \\ & & 0 & \dots & a'_{ll} \\ & & & \dots & * \end{bmatrix}$$

$$\Theta = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} := \begin{pmatrix} c & -s \\ s & c \end{pmatrix}$$

$$\begin{pmatrix} \alpha c^2 - 2\gamma cs + \beta s^2 & -\gamma s^2 + (\alpha - \beta)cs + \gamma c^2 \\ -\gamma s^2 + (\alpha - \beta)cs + \gamma c^2 & \beta c^2 + 2\gamma cs + \alpha s^2 \end{pmatrix} = \begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix}$$

$$Q_{kl} = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & c & \dots & s \\ & & \vdots & \ddots & \vdots \\ & & -s & \dots & c \\ 0 & & & & \ddots & \\ & & & & & & 1 \end{bmatrix}$$

Програмна реалізація

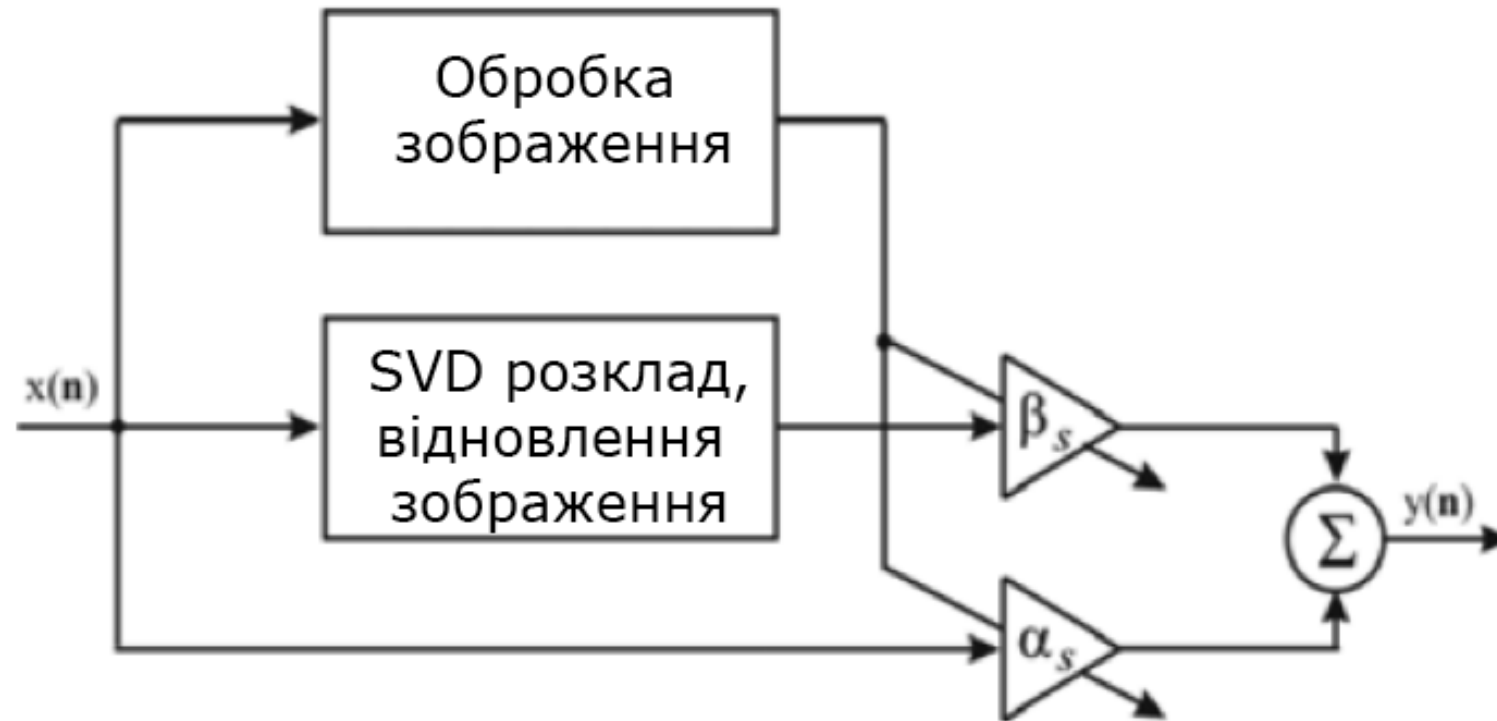
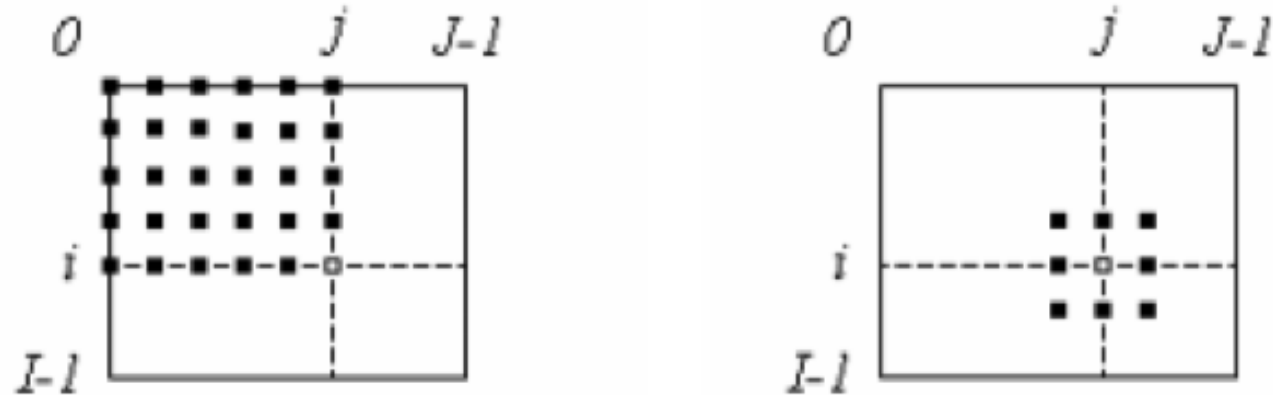


Схема роботи програмного продукту

Створення пошкодженого зображення

Необхідність перетворення кожної матриці за градаціями RGB



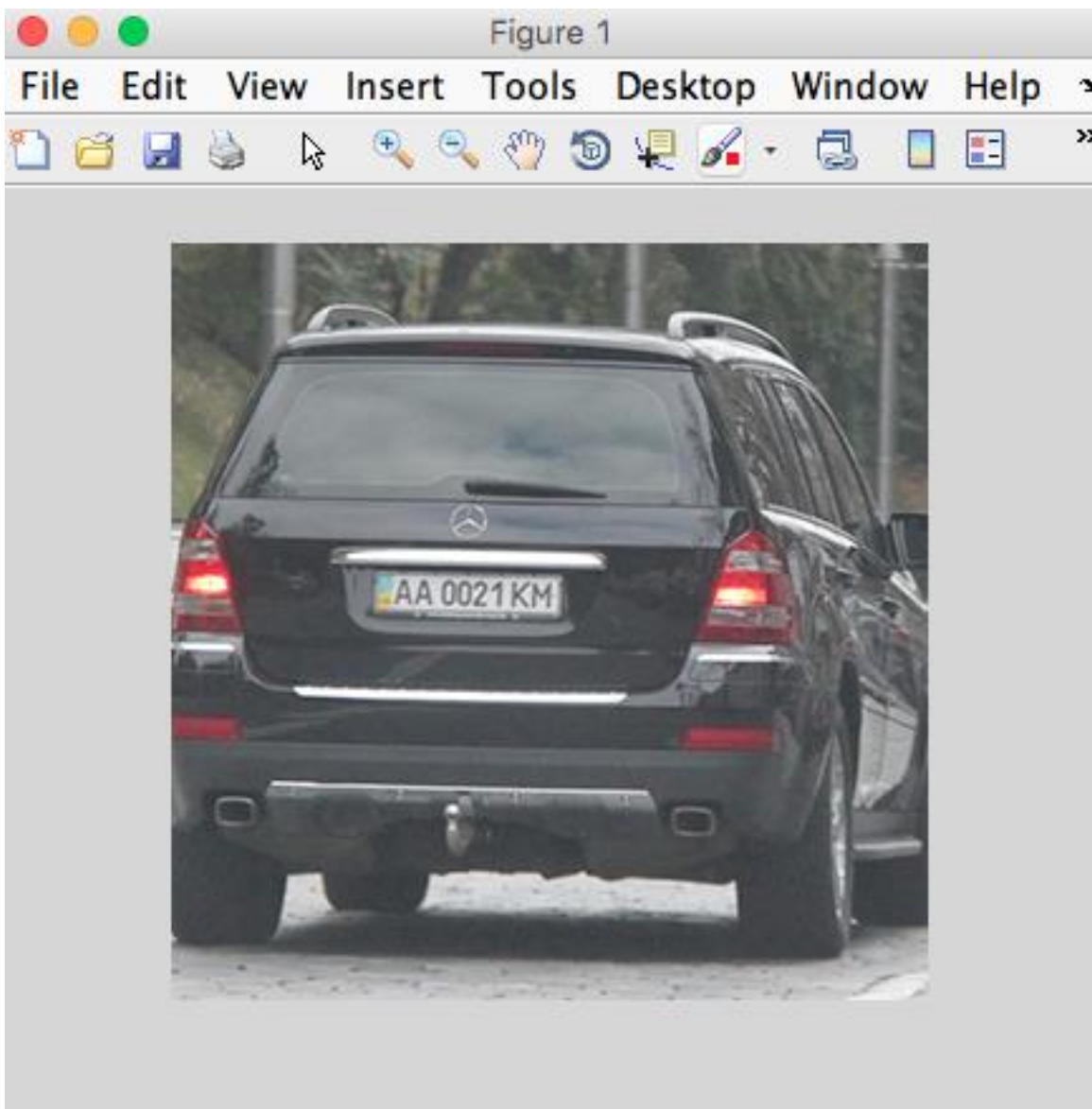
На прикладі околиць першого рівня

R11	R1n
..
..
..
Rm1	Rmn

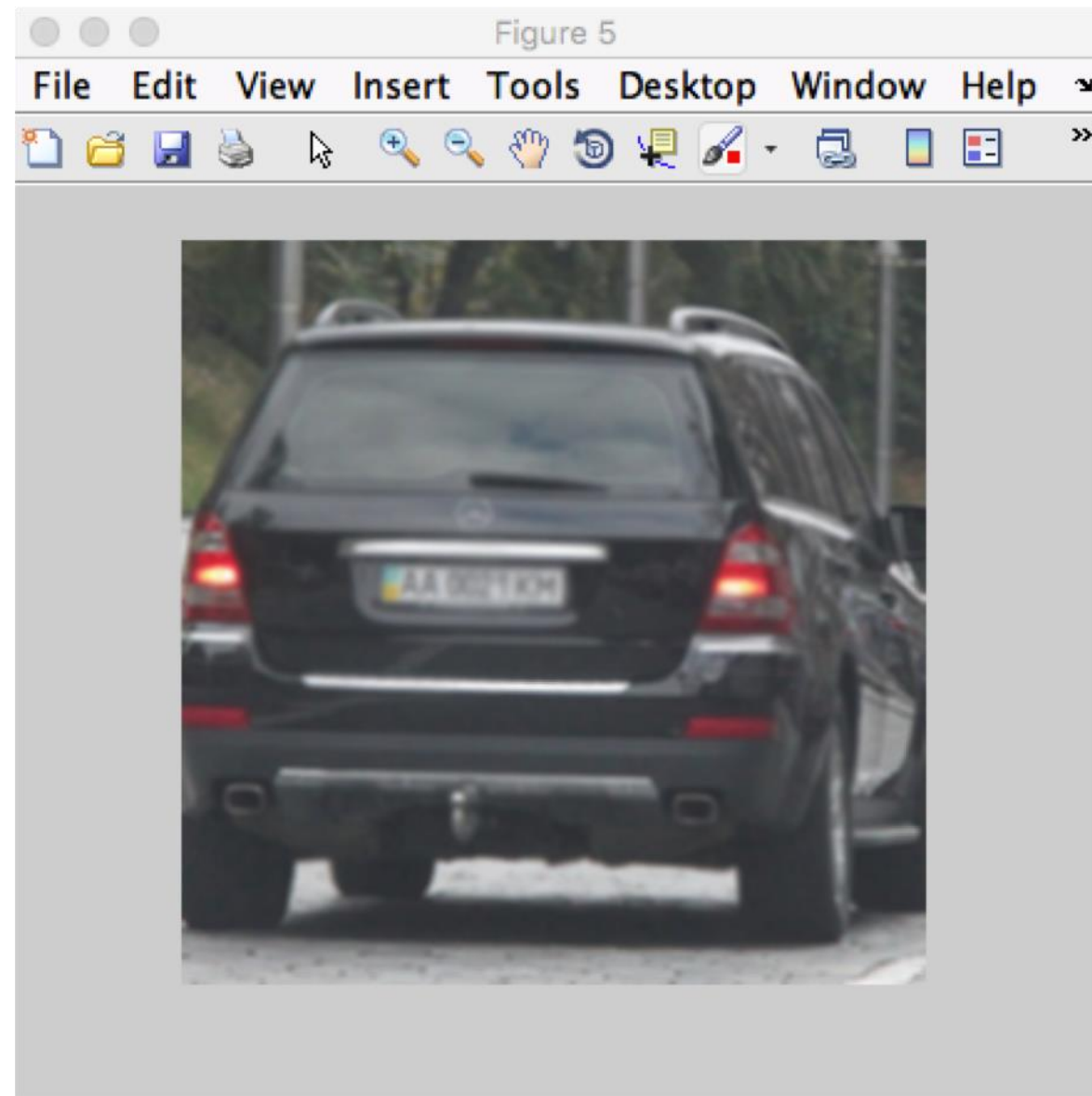
=>

$\Sigma/4$	$\Sigma/6$..	$\Sigma/4$
$\Sigma/6$	$\Sigma/9$
$\Sigma/6$	$\Sigma/9$
..
$\Sigma/4$	$\Sigma/4$

Результат розмиття



=>



Перетворення матриць

$\Sigma/4$	$\Sigma/6$..	$\Sigma/4$
$\Sigma/6$	$\Sigma/9$
$\Sigma/6$	$\Sigma/9$
..
$\Sigma/4$	$\Sigma/4$

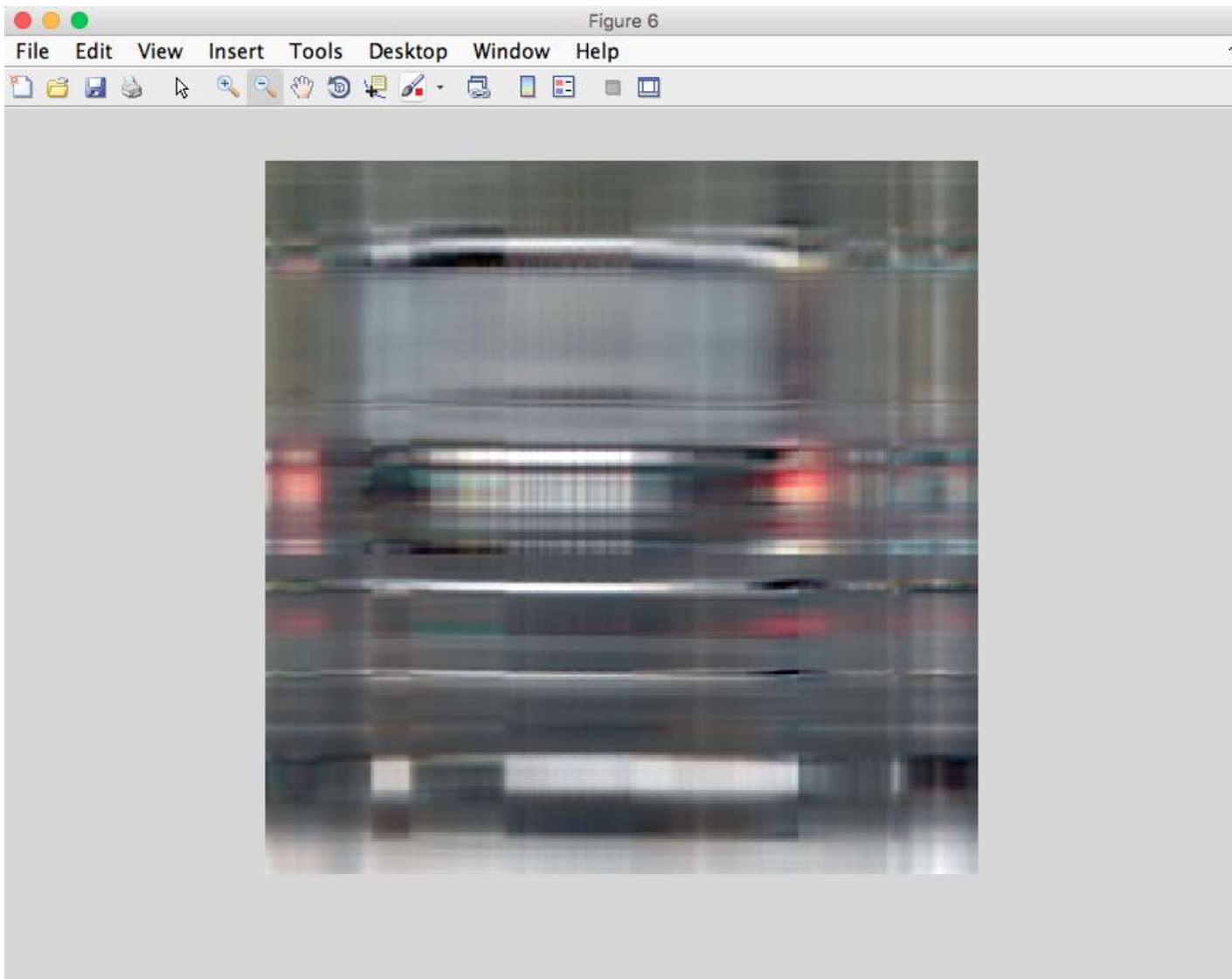
Зображення у матричній формі

=>

R1
..
..
..
Rnxm

Зображення у
формі стеку

Знаходження матриці пошкоджень

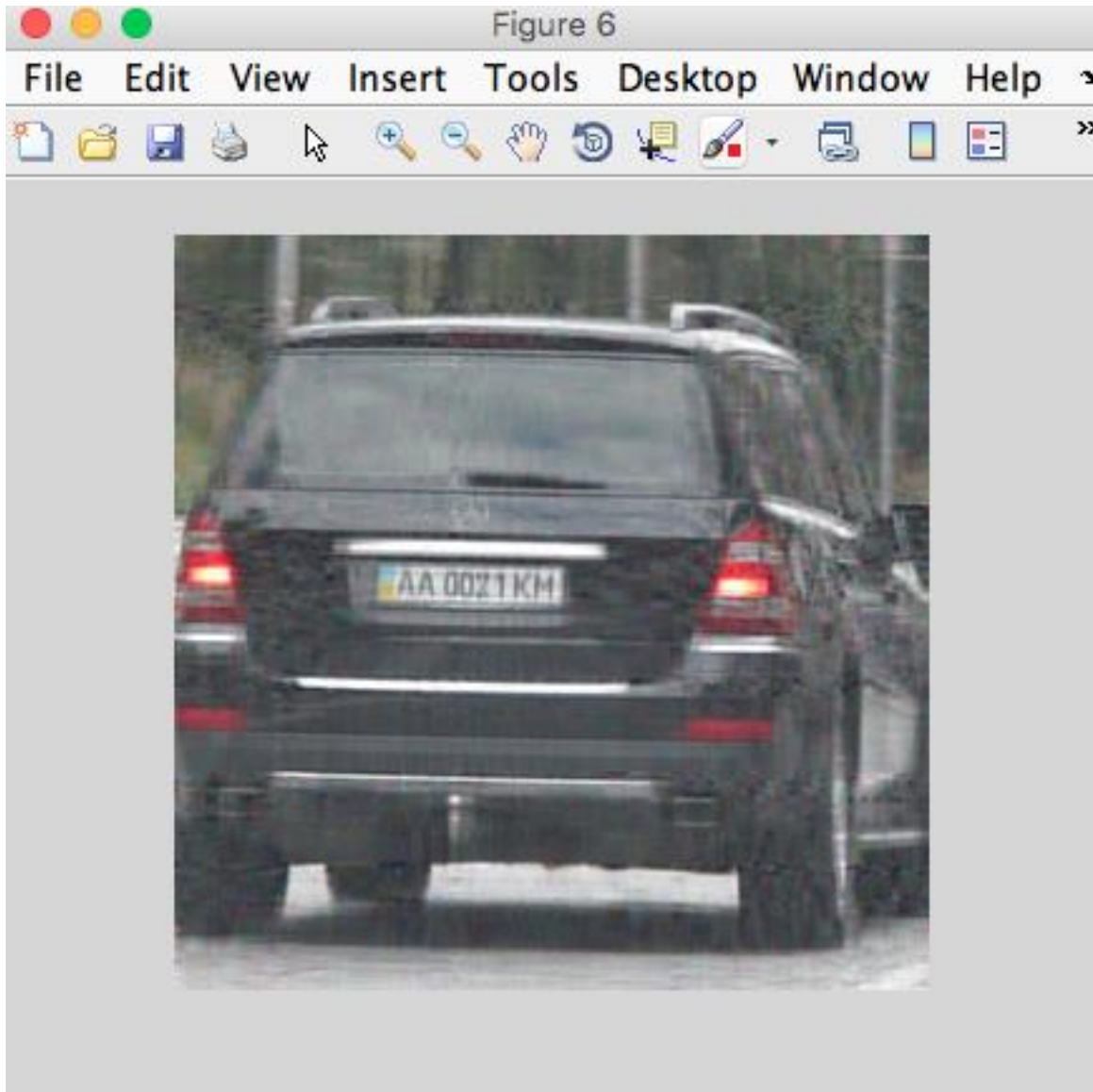


$$\begin{aligned}Ax &= y \\ A &= USV^t \\ USV^t x &= y \\ x &= V^t S^{-1} U y\end{aligned}$$

де:

A - матриця пошкоджень,
 x - початкове зображення, приведенне до форми стеку
 y - пошкоджене зображення, приведенне до форми стеку

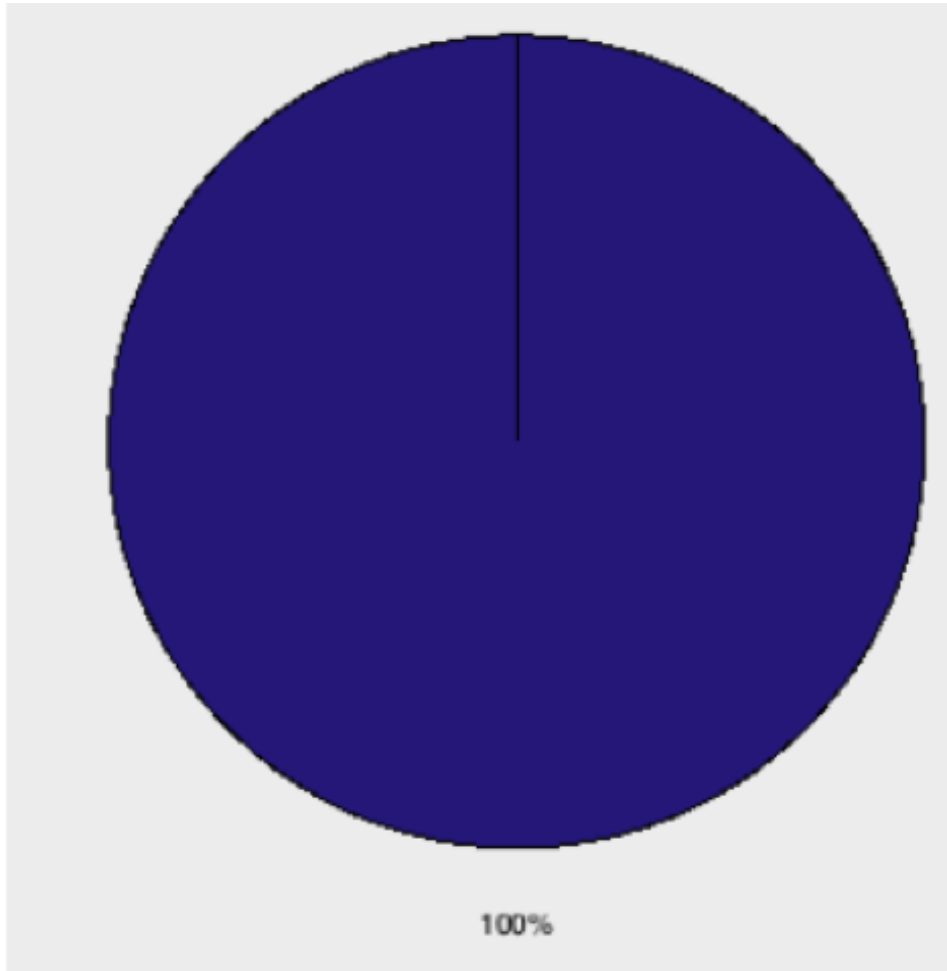
Відновлення зображення



$$x = V^t S^{-1} U y$$

$V^t S^{-1} U$ - матриця сингулярного розкладу пошкоджень,
x - початкове зображення, приведенне до форми стеку
y - пошкоджене зображення, приведенне до форми стеку

Аналіз швидкості роботи методу

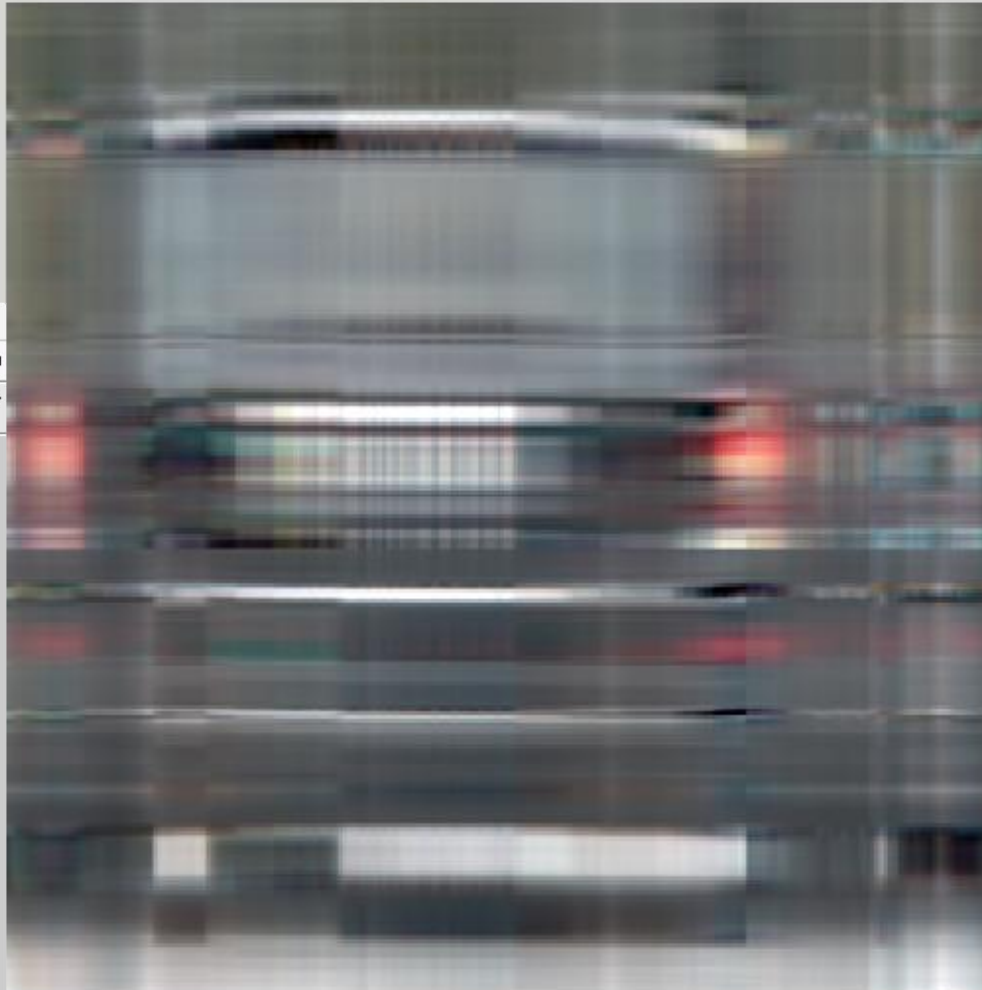
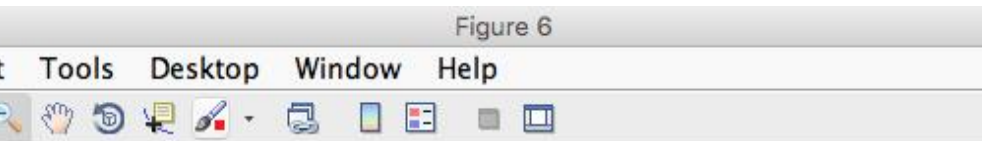
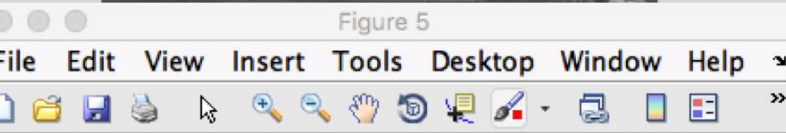


Стандартний SVD



Удосконалений SVD

Приклад роботи програми



Висновки

- Удосконалено метод відновлення зображень за рахунок застосування методу сингулярного розкладу з методом обертань Якобі, що призвело до значного прискорення та підвищення якості результатів роботи методу;
- набуло подальшого розвитку застосування алгоритмів SVD та обертань Якобі для виправлення пошкоджень кольорових зображень.

На основі запропонованих удосконалень алгоритмів відновлення кольорових зображень, була розроблена система, яка за рахунок використання удосконаленого методу усунення пошкоджень, по зручності користування та якості отриманих результатів перевершує існуючі аналоги.

Створена комп'ютерно-інформаційна система значно спрощує процес відновлення пошкоджених зображень. Розроблений програмний продукт може бути використаний, наприклад, в системах відновлення даних, відеоспостереженні, а також системах аналізу масивів графічної інформації

Дякую за увагу!