

# User Manual

## 1. Introduction

The Image Utility application is a console-based program written in C that allows users to manipulate BMP images. It supports basic editing functions such as rotation, scaling, resizing, and cropping, as well as steganography features for hiding and recovering messages within images.

This manual explains how to install and run the application, describes each feature in detail, and provides example workflows with console output and sample images.

---

## 2. System Requirements

### Supported Platforms

- **Linux:** Ubuntu 22.04 LTS (64-bit), GCC 11.4
- **Windows:** Windows 11 Pro (64-bit), Visual Studio 2022 Build Tools (MSVC 19.41)

### Supported File Formats

- 24-bit uncompressed BMP
- Some of 32-bit uncompressed BMP

Other formats (PNG, JPG, etc.) are not supported.

---

## 3. Installation and Setup

### Linux

1. Navigate to the project directory.
2. Build the application using:
3. `make`
4. Run the tool:
5. `bin/linux/imagetool`

### Windows

1. Open the **Developer Command Prompt for VS 2022** (x64 Native Tools).
2. Navigate to the project folder:
3. `cd "C:\Software Analysis & Debugging\hw01_mock"`

4. Build using:
5. `nmake /f Makefile.win`
6. Run: `bin\windows\imagetool.exe`

---

## 4. System Requirements

=== Image Utility ===

Commands:

<code>load &lt;filename&gt;</code>	- Load a BMP file
<code>save &lt;filename&gt;</code>	- Save current image
<code>fill &lt;R&gt; &lt;G&gt; &lt;B&gt;</code>	- Fill image with color
<code>rotate &lt;angle&gt;</code>	- Rotate by angle (degrees)
<code>scale &lt;factor&gt;</code>	- Scale by factor (e.g. 0.5, 2.0)
<code>resize &lt;w&gt; &lt;h&gt;</code>	- Resize to width/height
<code>crop &lt;x&gt; &lt;y&gt; &lt;w&gt; &lt;h&gt;</code>	- Crop region
<code>embed &lt;message&gt;</code>	- Hide text inside image
<code>extract</code>	- Recover hidden text from image
<code>exit</code>	- Quit program

---

## 5. Features

Command : `load <filename>`

On success : “Loaded <filename> <width>x<height> <bitCount>”

On failure : “Failed to load <filename>”

Supports any uncompressed BMPs

Command : `save <filename>`

On success : “Saved image to <filename>”

On failure : “No image loaded” if it is not loaded “Failed to save image” for other issues

Supports any typed of BMPs

Command: `fill <R> <G> <B>`

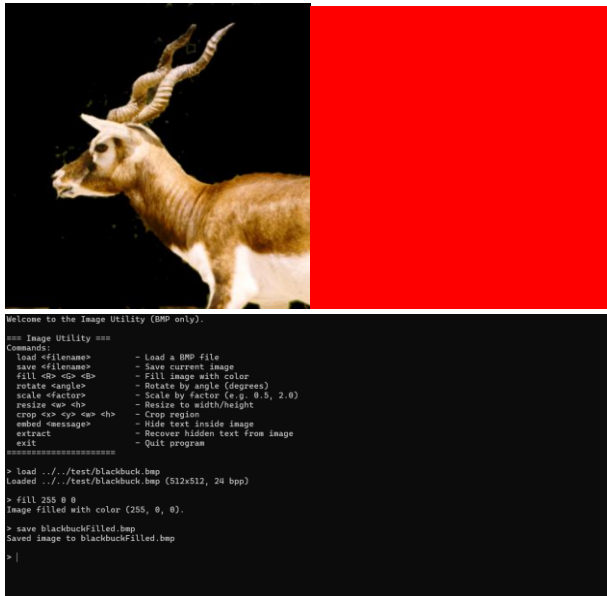
On success: Image filled with color (<R>, <G>, <B>).

If no image loaded: No image loaded.

If wrong arguments: Usage: `fill <R> <G> <B>`

Supports only 24-bit BMPs

Example:



Command: rotate <angle>

On success: Rotated image by <angle> degrees.

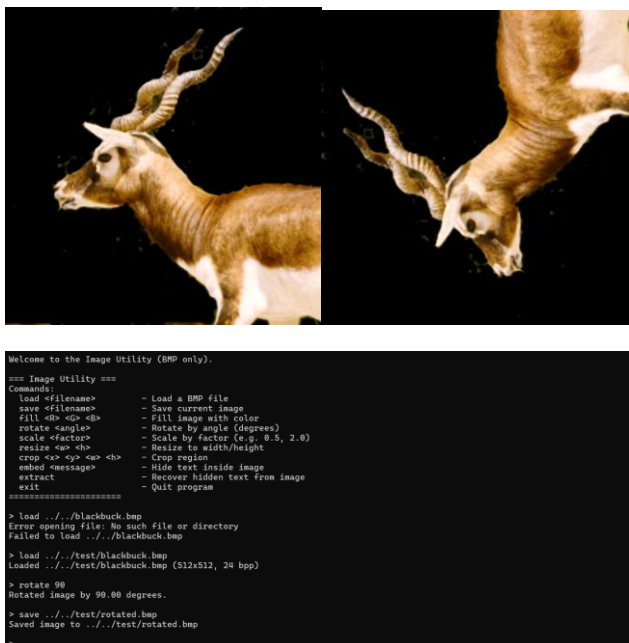
If no image loaded: No image loaded.

If missing argument: Usage: rotate <angle>

On failure: Rotation failed.

Supports only 24-bit BMP

Example:



Command: scale <factor>

On success: Scaled image by <factor> x.

If no image loaded: No image loaded.

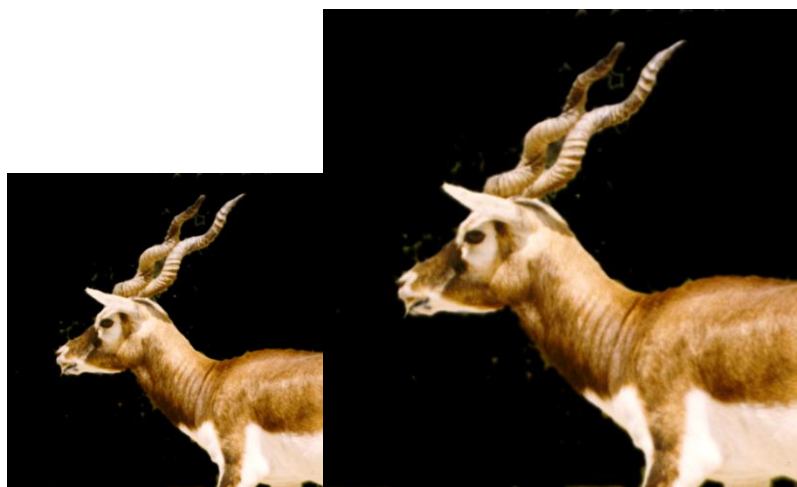
If missing argument: Usage: scale <factor>

If factor  $\leq 0$ : Scale factor must be  $> 0$ .

On failure: Scaling failed.

Supports only 24 bit BMP only

Example:



```
Welcome to the Image Utility (BMP only).

=== Image Utility ===
Commands:
  load <filename>      - Load a BMP file
  save <filename>      - Save current image
  fill <R> <G> <B>    - Fill image with color
  rotate <angle>       - Rotate by angle (degrees)
  scale <factor>       - Scale by factor (e.g. 0.5, 2.0)
  resize <w> <h>      - Resize to width/height
  crop <x> <y> <w> <h> - Crop region
  embed <message>      - Hide text inside image
  extract              - Recover hidden text from image
  exit                - Quit program

=====

> load ../../test/blackbuck.bmp
loaded ../../test/blackbuck.bmp (512x512, 24 bpp)

> scale 2
Scaled image by 2.00x.

> save ../../test/scaled.bmp
Saved image to ../../test/scaled.bmp

> |
```

Command: resize <width> <height>

On success: Resized image to <width> x <height>.

If no image loaded: No image loaded.

If missing arguments: Usage: resize <w> <h>

On failure: Resize failed.

Support: 24 and 32 bit BMPs

Example :



```

Welcome to the Image Utility (BMP only).
run Image Utility run
Commands:
load <filename>      - Load a BMP file
save <filename>      - Save current image
fill <cb> <cb> <cb>  - Fill image with color
rotate <angle>       - Rotate by angle (degrees)
scale <factor>       - Scale by factor (e.g. 0.5, 2.0)
resize <w> <h>       - Resize to width/height
crop <x> <y> <w> <h> - Crop region
embed <message>      - Hide text inside image
extract              - Recover hidden text from image
exit                 - Quit program
=====
> load ../test/blackback.bmp
loaded ../test/blackback.bmp (512x512, 24 bpp)

> resize 500 500
Resized image to 500x500.

> save ../test/resized.bmp
Saved image to ../test/resized.bmp
.

```

Command: crop <x> <y> <w> <h>

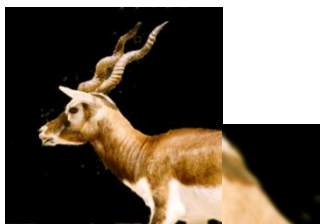
On success: Cropped to region (<x>, <y>, <w>, <h>).

If no image loaded: No image loaded.

On failure: Crop failed

Supports 24-bit and 32-bit BMPs

Example:



```

Welcome to the Image Utility (BMP only).
run Image Utility run
Commands:
load <filename>      - Load a BMP file
save <filename>      - Save current image
fill <cb> <cb> <cb>  - Fill image with color
rotate <angle>       - Rotate by angle (degrees)
scale <factor>       - Scale by factor (e.g. 0.5, 2.0)
resize <w> <h>       - Resize to width/height
crop <x> <y> <w> <h> - Crop region
embed <message>      - Hide text inside image
extract              - Recover hidden text from image
exit                 - Quit program
=====
> load blackback.bmp
Error opening file! No such file or directory
Failed to load blackback.bmp

> load ../test/blackback.bmp
loaded ../test/blackback.bmp (512x512, 24 bpp)

> crop 250 250 50 50
Cropped to region (250,250,50,50).

> save ../test/cropped.bmp
Saved image to ../test/cropped.bmp
.

```

Command: embed <message>

On success: Message embedded

IF missing argument: Usage: embed <message>

If no image loaded: No image loaded.

On failure: Failed to embed message.

Supports 24-bit and 32-bit

Example:

```
Welcome to the Image Utility (BMP only).

=== Image Utility ===
Commands:
  load <filename>      - Load a BMP file
  save <filename>      - Save current image
  fill <cr> <G> <B>   - Fill image with color
  rotate <angle>       - Rotate by angle (degrees)
  scale <factor>        - Scale by factor (e.g. 0.5, 2.0)
  resize <w> <h>        - Resize to width/height
  crop <x> <y> <w> <h> - Crop region
  embed <message>       - Hide text inside image
  extract              - Recover hidden text from image
  exit                 - Quit program
=====

> load ../../test/blackbuck.bmp
Loaded ../../test/blackbuck.bmp (512x512, 24 bpp)

> embed Hello from the picture!
Message embedded.

> save ../../test/embed.bmp
Saved image to ../../test/embed.bmp

> |
```

Command: extract

On success: Extracted message: <message>

If no image loaded: No image loaded.

On failure: No message found

```
Commands:
  load <filename>      - Load a BMP file
  save <filename>      - Save current image
  fill <cr> <G> <B>   - Fill image with color
  rotate <angle>       - Rotate by angle (degrees)
  scale <factor>        - Scale by factor (e.g. 0.5, 2.0)
  resize <w> <h>        - Resize to width/height
  crop <x> <y> <w> <h> - Crop region
  embed <message>       - Hide text inside image
  extract              - Recover hidden text from image
  exit                 - Quit program
=====

> load ../../test/blackbuck.bmp
Loaded ../../test/blackbuck.bmp (512x512, 24 bpp)

> embed Hello from the picture!
Message embedded.

> save ../../test/embed.bmp
Saved image to ../../test/embed.bmp

> load ../../test/embed.bmp
Loaded ../../test/embed.bmp (512x512, 24 bpp)

> extract
Extracted length = 23
Extracted message: "Hello from the picture!"

> |
```

Command: exit

Exits the program

---

## 6. Error Messages and Troubleshooting

- **No image loaded.** → You must load a BMP before using operations.
  - **Failed to load <filename>.** → File not found or unsupported format.
  - **Scale factor must be > 0.** → Use a positive scaling factor.
  - **Crop rectangle out of bounds.** → Chosen rectangle is outside the image.
  - **Failed to embed message.** → The hidden message is too long.
- 

## 7. Known Limitations

- Only uncompressed 24-bit and 32-bit BMPs are supported.
- Rotation uses simple interpolation and may produce jagged edges.
- Scaling uses nearest-neighbor (fast but less smooth).