LAB 6:
Automatical counting of blod cells

Computer Vision Laboratory
Linköping University, Sweden

0 Preparations, introduction and start of the lab system

In this computer exercise, you will perform experiments with different correlation methods. You will also implement a thresholding method with hysteres and finally solve the task of automatically counting blood cells.

☞ Read through this paper, repeat related theory and perform the home exercise (marked with a pointing hand) before the session.

Log in and open a terminal window from the background of the screen. Then give the following commands in the terminal window:

```
module add matlab
matlab
```

Then give the following commands in the MATLAB window:

```
initcourse('TSBB08');
mips;
```

The command initcourse('TSBB08') sets up correct paths in MATLAB so that the desired files can be found.

1 Pattern Recognition with template matching

1.1 Common correlation

The common correlation (or actually cross-correlation) can be used to detect the presence of a pattern $m(x, y)$ in an image $b(x, y)$. The result of the correlation is given by

$$(m\Box b)(x, y) = \sum_{\alpha}\sum_{\beta} m(\alpha, \beta) \cdot b(x + \alpha, y + \beta).$$

Here, we use the square to denote correlation. Try this method to locate a text-patch in the image **clic**. Use the function **chooseobject** to choose the patch. By writing the following MIPS-command in the MATLAB window, a 24×24-box at the coordinate $(16, 30)$ is chosen:

```
execute('chooseobject', 'A1', 16, 30, 24);
```

Select the function **corr** in the **Operation window**. Then threshold the correlation image with **threshold**. If you like, you can first look for a suitable threshold by using the color table **Interactive**. Why does this template matching method not work here?

1.2 Normalized correlation

Normalized correlation, in our system called **corrn**, is given by

$$\begin{aligned}(m\Box_n b)(x,y) &= \frac{(m\Box b)(x,y)}{\|m(x,y)\| \cdot \|b(x,y)\|_{\text{neighborhood}}} \\ &= \frac{\sum_\alpha \sum_\beta m(\alpha,\beta) \cdot b(x+\alpha, y+\beta)}{\sqrt{\sum\sum_{\alpha,\beta} m^2(\alpha,\beta) \cdot \sum\sum_{\alpha,\beta\in\text{neighborhood}} b^2(x+\alpha, x+\beta)}}\end{aligned}$$

Here we divide with the norm of the pattern and the image in the actual environment. In this way we avoid high values of the correlation result because of high image values. The maximum value of $(m\Box b)(x,y)$ is 1 and it is obtained only when there is a perfect match between pattern and image. (In MIPS, the maximum value is 255 because $(m\Box b)(x,y)$ is multiplied with 255.) Try **corrn** on the previous example. Result?

1.3 Correlation without local DC-level

Correlation without local average, in our system called **corrdcp**, is given by

$$(m\Box b)_{DC}(x,y) = \sum_\alpha \sum_\beta [m(\alpha,\beta) - \bar{m}] \cdot [b(x+\alpha, y+\beta) - \bar{b}].$$

The local mean in the image, $\bar{b}$, is calculated in a neighborhood, whose size is determined by the size of the pattern. This prevents high correlation values due to high image values. Try **corrdcp** on the previous example. Result?

1.4 Normalized correlation without local DC-level, i.e. the correlation coefficient

Normalized correlation without local DC-level i.e. the correlation coefficient, is called **corrc** in MIPS and is given by

$$(m\Box_c b)(x,y) = \frac{\sum_\alpha \sum_\beta [m(\alpha,\beta) - \bar{m}] \cdot [b(x+\alpha, y+\beta) - \bar{b}]}{\sqrt{\sum\sum_{\alpha,\beta}[m(\alpha,\beta) - \bar{m}]^2 \cdot \sum\sum_{\alpha,\beta\in\text{neighborhood}}[b(x+\alpha, x+\beta) - \bar{b}]^2}}$$

Try **corrc** on the previous example. Result?

In the previous examples, the pattern matched perfectly with the image. In a real case, the consistency is usually not perfect. Load the image **blod256**, cut out one blood cell and test **corr**, **corrn**, **corrdcp** and **corrc** again. Result?

2 Thresholding with hysteres

In the lecture notes, thresholding with hysteres was described. Sketch on a MATLAB program for thresholding with hysteres.

Provide a paper with your program!

Write your MATLAB program to a file and experiment with it until it works. Try your implementation on **nuf0b**. Do you get a similar result as shown in the image below?

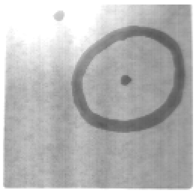
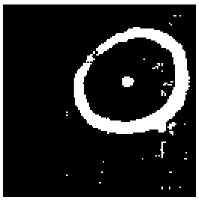
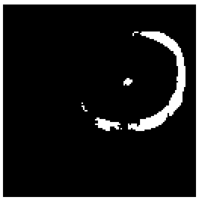


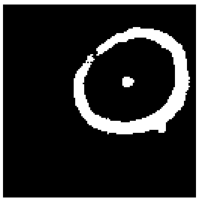
image nuf0b



after threshold1



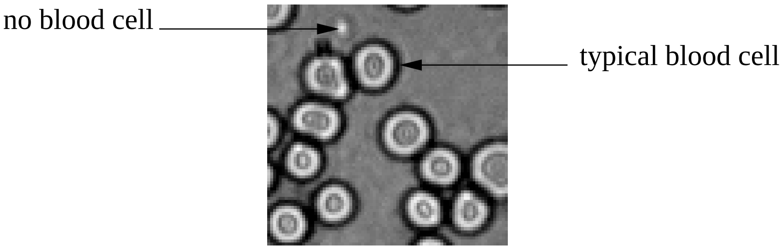
after threshold2



resulted image

3 Microscopy

Microscopy images are very well suited for two-dimensional image processing since these images are in principle two-dimensional. You will here design a MATLAB program that counts the number of red blood cells in an image. A section of a typical blood cell image is shown below. You can use the images **blod256** or **blod512**.



A problem that may arise is that one blood cell can be split into two and therefore be counted as two. Another possible problem is that small objects that are not blood cells may be counted as well. This problem may be handled by removing too small objects, however. These are some of the problems you should be able to handle. Solve the problem in two ways, first based on correlation and secondly based on thresholding with hysteres!

When using the latter it is recommended to focus on the bright rings instead of the dark rings. Consequently, the type of threshold should be '>' instead of '<' as in the previous section.

Which of the two programs work best?

Discuss which blood cells the programs found, which blood cells the programs missed, and why.

____

____

4 Hint for MATLAB programming

You will probably need the following MATLAB commands: **while**, **for** and **sum (sum (image))**. Type **help** and the command name to get more information, for example: **help sum**. Some of you may use the MIPS-commands **labelhistogram** and **selectobj**. You will also need some of the MIPS commands from previous computer exercises.