

## ABOUT

KoalaTools is a small collection of utility functions to manipulate koala pictures. It can move, delete, combine pictures and do stuff like using a picture as “mask” for another one. Moreover, it can also show you some information about the koala picture and so forth.

## USAGE

```
KoalaTool info [<col> <row>] <src>
           del <x1> <y1> <x2> <y2> <src> [<dst>]
           crop <x1> <y1> <x2> <y2> <src> [<dst>]
           delv <from> [<to> [<step>]] <src> [<dst>]
           filter <src> [<dst>]
           move <charsh> <charsv> <src> [<dst>]
           move(v|h) <chars> <src> [<dst>]
           insert <x1> <y1> <x2> <y2> <pic1> <pic2> [<dst>]
           insert(v|h) <chars> <pic1> <pic2> [<dst>]
           replace <x> <y> <x2> <y2> <pic1> <pic2> [<dst>]
           replace(v|h) <chars> <pic1> <pic2> [<dst>]
           merge <pic1> <pic2> [<dst>]
           mask|imask <src> <mask> [<dst>]
           examples
           help
```

## DESCRIPTION

```
info ..... Show information about the picture [and a specified block/char]
del ..... Deletes a rectangle out of the picture (unit is chars)
crop ..... Crops a rectangular part out of the picture (unit in chars)
<x1y1x2y2> ... Coordinates presenting a character position (x: 0-39, y: 0-24)
delv ..... Delete lines <from> to <to> (unit is pixels)
<from> ..... Starting line (0-199 pixels)
<to> ..... Last line (0-199 pixels), Default is 199
<step> ..... Delete each n-th line (default is 1 which means each line)
filter ..... Removes every second koala (=double) pixel within the picture
move ..... Move picture horizontally and vertically.
<charsh> ..... Negative number moves upwards, positiv one downwards (-24 to 24)
<charsv> ..... Negative number moves left, positiv right (-39 to 39 chars)
move(v|h)..... Move picture horizontally / vertically
insert ..... Inserts pic2 into pic1 at the given rectangle
               Insertion of pic2 starts at top left corner (coordinate 0/0)
insert(v|h) .. Inserts/appends pic2 after pic1 horizontally / vertically
<chars> ..... First row where 2nd pic starts (1-24 chars)
replace ..... Replace rectangular region of pic 1 by same region of pic 2.
replace(v|h) . Replace pic 1 by pic 2 vertically / horizontally.
<chars> ..... First row (1-39 chars) / line (1-24) where 2nd pic is used.
merge ..... Merges pic 2 into pic 1 (merge is perfromred grid by grid)
merge+ ..... Same as merge (Also uses pixels from pic 1 one on empty lines)
               (This will likely cause artifacts)
mask ..... Mask (leave pixel only if also a pixel in the mask exists)
imask ..... Invert mask (leave pixel only if no pixel in the mask exists)
<src> ..... Source Koala file
<mask> ..... Mask Koala file
<dest> ..... Destination Koala file
examples ..... Show examples (with no other options)
help ..... Show detailed help
```

Always provide the full file name including extension

## EXAMPLES

Koalatool info pic1.kla

Shows information about the picture.

```
C:\WINDOWS\system32\cmd.exe

C:\temp\Koala>koalatool info pic1.kla
Info:
Filename: 'pic1.kla'
Filesize: 10003
Startaddr: $6000
Background: 12 [grey]

C:\temp\Koala>
```

koalatool info 20 6 pic1.kla

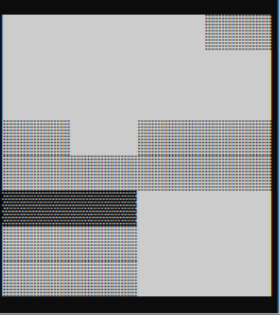
Show information about the file pic1.kla and coordinate 20/4.

```
C:\WINDOWS\system32\cmd.exe - run

D:\Koala>koalatool info 20 6 pic1.kla
Info:
Filename: 'pic1.kla'
Filesize: 10003
Startaddr: $6000
Background: 11 [dark grey]

Info of col 20, row 6:
Col-1 3 [cyan]
Col-2 13 [light green]
Col-3 14 [light blue]

fe 11111110
ff 11111111
ff 11111111
ba 10111010
aa 10101010
5f 01011111
af 10101111
af 10101111
```



koalatool del 10 5 28 10 pic1.kla

Deletes within the coordinates from 10/5 to 28/10.



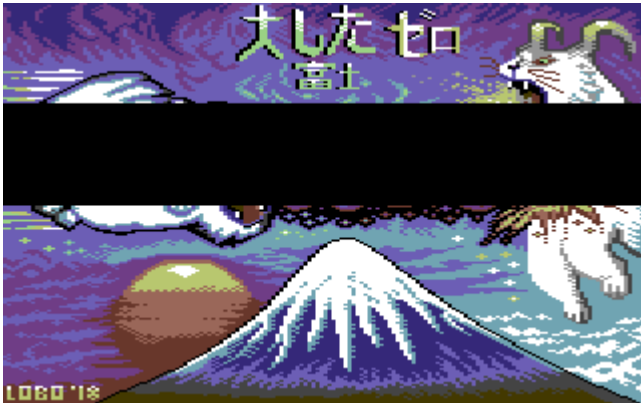
```
koalatool crop 10 5 28 10 pic1.kla
```

Crops within the coordinates from 10/5 to 28/10.



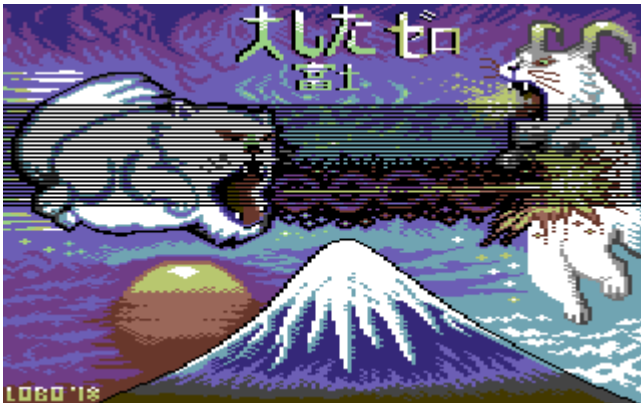
```
koalatool delv 50 100 pic1.kla
```

Deletes lines starting from line 50 to 100



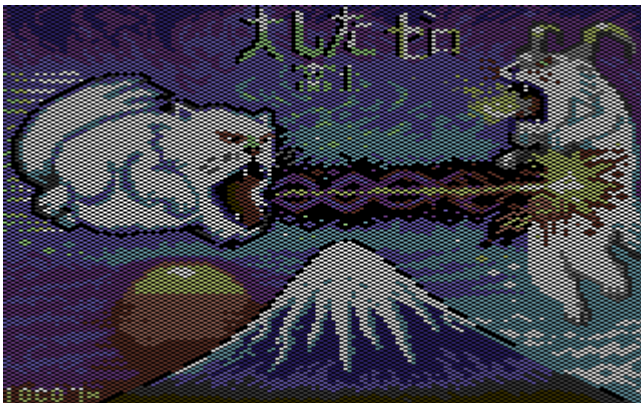
```
koalatool delv 50 100 2 pic1.kla
```

Deletes every 2<sup>nd</sup> line starting from line 50 to 100



```
koalatool filter pic1.kla
```

Use filter on pic.kla. Each second koala pixel is deleted.



```
koalatool move 5 -2 pic1.kla
```

Move pic 5 chars to the right and 2 chars up.



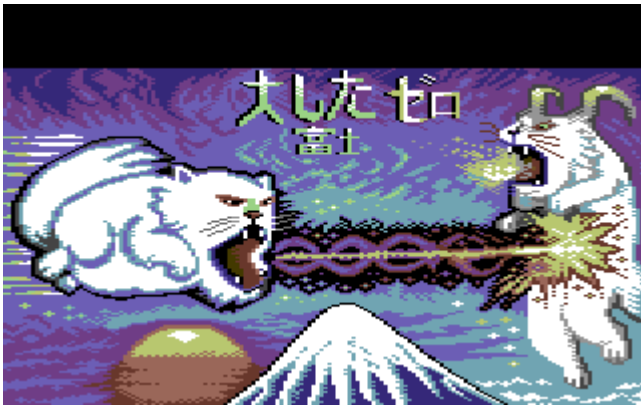
```
koalatool movev -3 pic1.kla
```

Move pic up 3 lines.



```
koalatool movev 4 pic1.kla
```

Move pic down 4 lines.



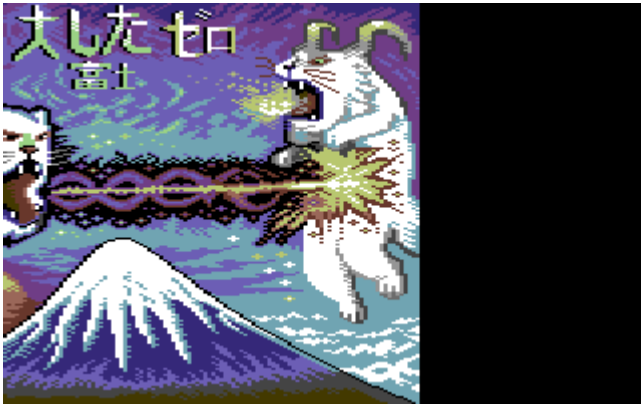
```
koalatool moveh 12 pic1.kla
```

Move pic 12 characters to the right.



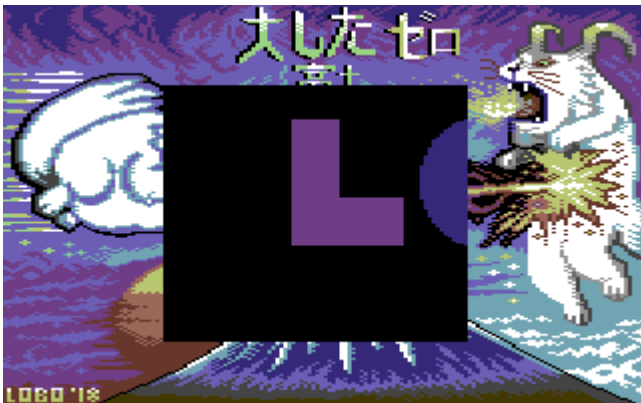
```
koalatool moveh -14 pic1.kla
```

Move pic 14 characters to the left.



```
koalatool insert 10 5 28 20 pic1.kla pic2.kla
```

Inserts pic2 into pic1 within the defined rectangle/frame.



```
koalatool insertv 6 pic1.kla pic2.kla
```

Joins pic2 to pic1 starting at char line 6.



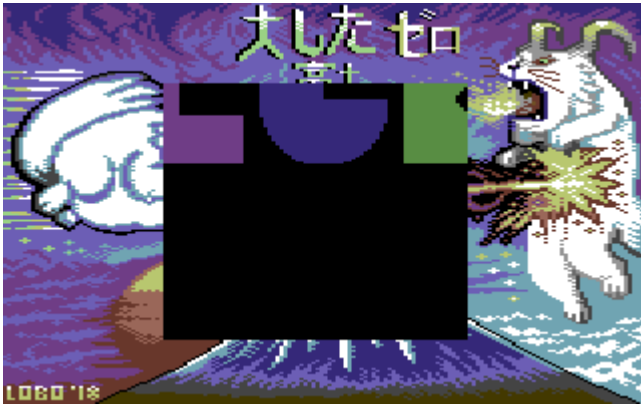
```
koalatool inserth 12 pic1.kla pic2.kla
```

Joins pic2 to pic1 starting at char column 12.



```
koalatool replace 10 5 28 20 pic1.kla pic2.kla
```

Replace the defined rectangle/frame from pic1 with the same rectangle/frame from pic2.



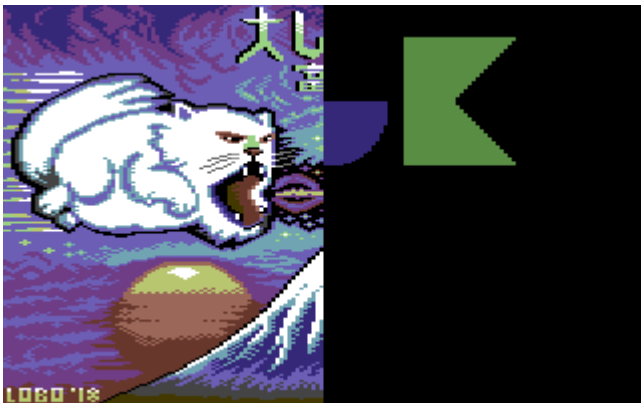
```
koalatool replacev 4 pic1.kla pic2.kla
```

Replace pic 1 with pic 2 starting after char (line) 4.



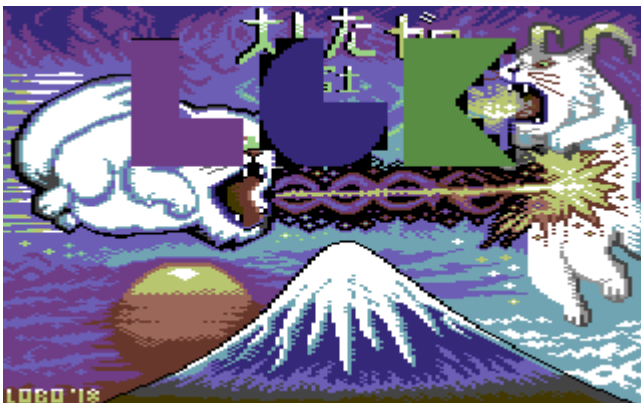
```
koalatool replaceh 20 pic1.kla pic2.kla
```

Shows left half of pic1 and right half of pic2.



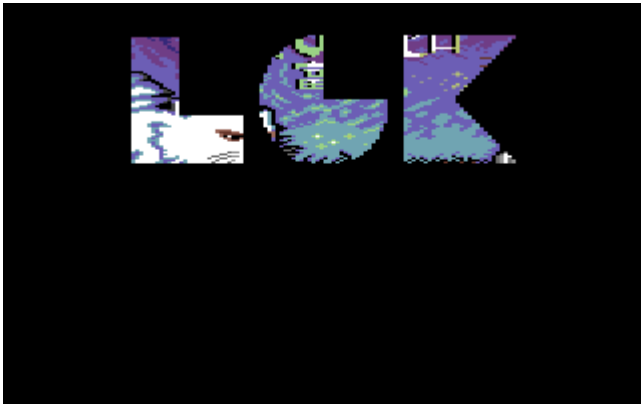
```
koalatool merge pic1.kla pic2.kla
```

Puts pic2 over pic1.



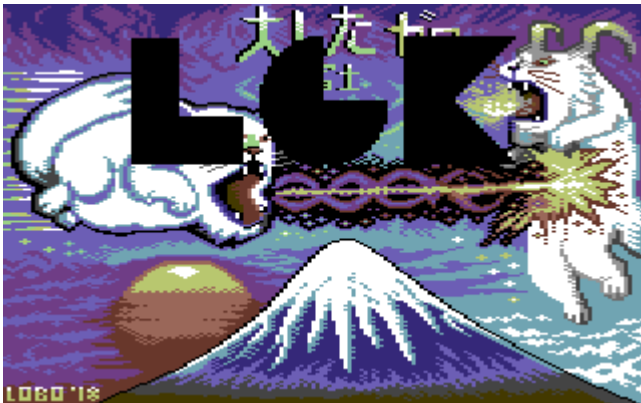
```
koalatool mask pic1.kla pic2.kla
```

Applies mask (pic2.kla) to picture (pic1.kla)



```
koalatool imask pic1.kla pic2.kla
```

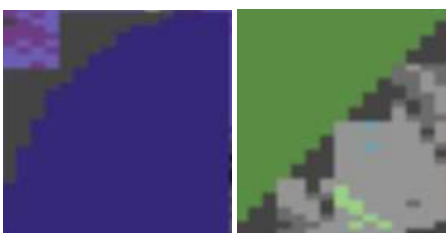
Applies mask (pic2.kla) to picture (pic1.kla) inverted (cut out)



#### NOTES ABOUT MERGE AND MERGE+

Merge and merge+ work on a block (=character) basis. A complete block is merged if it contains data. If it contains no data, it is not merged. So, in order to merge a block, it must have at least one pixel set.

A perfect picture would be aligned around the characters respectively by the 8x8 grid which is defined by these characters. If this is not the case, you will get artifacts like seen here:

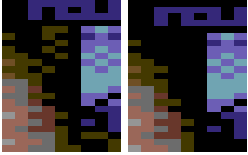


Merge+ will try to help out by at least filling empty lines within a character block. So, if the line (8 pixels, 4 koala pixels) are empty, it will take the pixels from the source and not the mask.



Within this example merge+ does not help a lot. Firstly, there is nearly no lines that are empty. Secondly, the colors are totally different so that the artifacts get even worse.

The following example is better, as more pixels could have been saved. But again, there is still one disadvantage remaining because the colors of both pictures do not match.



Because each block can use different colors a perfect solution does not exist. In most cases merge is the best option which this tool offers. In some few cases, merge+ might offer some benefits though.

#### OTHERS

Picture (Great Fuji Zero) used with permission of Lobo (<https://www.spitoufs.com/>).

Send feedback, questions, and requests to [admin@logiker.com](mailto:admin@logiker.com).

You can also take a look at my homepage: <https://logiker.com>.