

V language

Bowen Liang (bl2434) Language Guru
Peng Jiang (pj2243) System Integrator
Rongxin Du (rd2537) System Tester
Yun Zhang (yz2406) Project Manager

Mentor: Shuai Sun (ss4088)

V operations on Video

Operator		~	[:]	%>
Usage	join	fade in/out	cut in time	clip in space
Operator	=>	*>	**	@
Usage	change resolution	zoom in/out	image to video	capture image or pixel

Video effect sample code

```
video v1<<"source1.avi", v2<<"source2.avi",s,r;  
video q;  
r=2~v1~2;  
r>>"output_fade.avi";  
s = v1|v2;  
s>>"output_join.avi";  
q=v1=>{300,400};  
q>>"output_resize.avi";
```

Back-end Implementation

Three classes:

1. CvPixel

channel r , g, b

2. CvImage

read/save, resize, get metadata, grab pixel

3. CvVideo

read/write, get metadata, grab frame, fade,
zoom, crop, cut, join

Every operation on video will generate a new video file on disk (cvCreateVideoWriter)

Test case Design

White-box testing

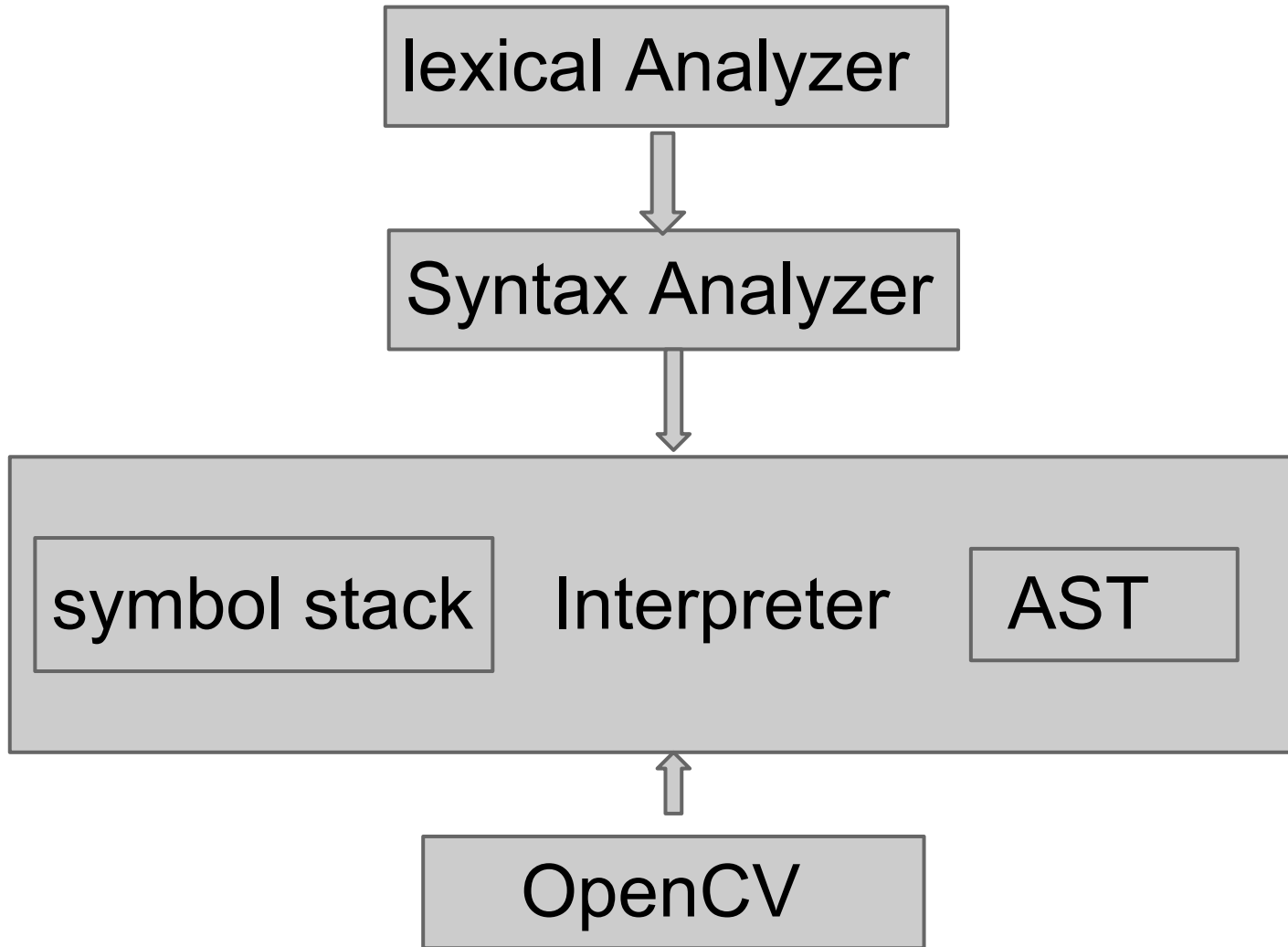
- based on three classes: video, image, pixel
- 9-possible mappings

Boundary Value Testing

- could only happen in the cut, zoom and crop operations on video files.
-

Unit testing plan

System Architecture



Highlights

- Compound Types
 - Arrays of types
 - `float[2][2] a = {{1,2},{3,4}}`
 - `float[][] b = a;`
 - `video[10] v << "source.avi";`
 - Function of types
 - `def mySqr : int <- (int k) return k * k;`
 - `int(int) sqr = mySqr;`
 - Parameter Passing
- Initially designed to support complicated user-defined transforms.

Function - example

- `def f : int <- (int k, int(int,int) sqr)
 return sqr(k);`
- `def mySqr : int <- (int k,int t = 0)
 return k*k+t;`
- `int(int, int) g = mySqr;`
- `f(1, g);`

Error Checking

- Assignment compatibility
- Operation compatibility
 - Numerical operations
 - Video manipulation
- Variable redeclaration
- Variable not defined
- Divide zero error

Future works

- Error Checking need to be further improved.
- builded-in function need to be added as a library, such as `overlap`, `super_overlap`
- may also rewrite it as a compiler and generate certain target code if possible