OpenCV - Feature # 1455: NEON optimisation of cv::threshold() for iOS

Status:	Done	**Priority:**	Normal
Author:	Yasuhiro Yoshimura	**Category:**	imgproc, video
Created:	2011-10-31	**Assignee:**	Vadim Pisarevsky
Updated:	2011-11-01	**Due date:**	
Difficulty:			
Pull request:			
Description:	I implemented NEON optimisation of cv::threshold() for iOS. I checked that patched cv::threshold() become fast about 10x speed up on the device(iPod touch 4th). Maybe, I think that this patch is also effective on Android.		

Associated revisions

2013-09-17 11:18 am - Alexander Smorkalov

Merge pull request #1455 from ilya-lavrenov:ocl_test_output

History

2011-10-31 05:48 pm - Andrey Kamaev

- Status changed from Open to Done

- (deleted custom field) set to invalid

This code can not be included into the [[OpenCV]], because it can fail with SIGSEG after attempt to write unallocated memory.

2011-11-01 02:24 pm - Yasuhiro Yoshimura

Replying to [comment:1 andrey.kamaev]:
> This code can not be included into the [[OpenCV]], because it can fail with SIGSEG after attempt to write unallocated memory.

Thank you for your comment.
I understand. I should add the following processing at
the beginning of this function.

```
if( _src.empty() || _dst.empty() )
{
    return;
}
```

But, if "src" and "dst" Mat are NULL, roi.width and roi.width are initialized to 0.
So, unallocated memory in not accessed.

2011-11-01 02:49 pm - Andrey Kamaev

Replying to [comment:2 dandelion]:
> Replying to [comment:1 andrey.kamaev]:
> > This code can not be included into the [[OpenCV]], because it can fail with SIGSEG after attempt to write unallocated memory.
>
> Thank you for your comment.
> I understand. I should add the following processing at
> the beginning of this function.
>
> if(_src.empty() |@@| _dst.empty())

> {

> return;

> }

>

> But, if "src" and "dst" Mat are NULL, roi.width and roi.width are initialized to 0.

> So, unallocated memory in not accessed.

Empty Mats is not a real problem of your code. You are wrong in the leftovers processing (all the cycles making @j+=8@). Also I should note that copying SSE optimization in NEON intrinsics rarely result in good code and I think that you can make a noticeably faster version using more suitable instructions.

Files

opencv231_neon_thresh_8u.patch	6.3 kB	2011-10-31	Yasuhiro Yoshimura