

---

Subject: Re: Annular Zernike polynomials

Posted by on Fri, 15 Mar 2013 23:28:41 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Den fredagen den 15:e mars 2013 kl. 17:56:07 UTC+1 skrev kagol...@lbl.gov:

> I have a lot of experience with this issue: Zernike polynomials on all types of pupils.

>

> If you use textbook definitions of the polynomials, you quickly find that they are not orthogonal on a discrete domain. That means if you care about accuracy, \*any time\* you pixellate a circle, you should perform a Gram-Schmidt operation to produce a new basis that is very much like the analytic Zernike basis, but contains subtle differences that make it orthogonal. Here is a reference for a \*stable\* Gram-Schmidt routine

>

> [http://en.wikipedia.org/wiki/GramSchmidt\\_process#Algorithm](http://en.wikipedia.org/wiki/GramSchmidt_process#Algorithm)

>

> Fortunately, the domain you choose is entirely arbitrary. So this process allows you to insert your discrete, annular domain, instead of a discrete circular pupil, and it works in the same way.

>

> The coefficients you find during the Gram-Schmidt process are the elements of the transformation matrix between the analytic Zernike polynomials, and the ones that are orthogonal on your discrete domain, whatever domain you chose. That transformation matrix can be stored and reused, for example, and it's good for the domain you specified.

>

> I have used this approach for fitting on domains with central obscurations, spiders, contamination: basically any pupil 'mask' array. Just be careful that your pupil mask does not refer to points outside of the unit circle, where it's not appropriate to use the Zernike basis.

>

> If you do use textbook analytic Zernikes (Mahajan, etc.) you'll find that your functions won't be orthogonal, and they won't be balanced.

Right, I went ahead and implemented them myself based on a paper by Mahajan. Sort of a partial Gram-Schmidt, where you skip doing inner products with respect to modes that could not (theoretically) contribute. For my application I don't care much if the modes are exactly orthogonal, so that's good enough.

---