

A Simple Suggestion based on Understanding & Using a Combination of Python based Tools in the Context of Rabin Fingerprint Computing.

[Exploring - Medical Imaging/Electron Microscopy Imaging R&D]

Nirmal Tej Kumar

Senior Researcher Informatics/Imaging/AI/Photonics/Nanotechnology/HPC R&D.
R&D Collaborator USA/UK/Israel/BRICS Group of Nations.
Current Member ante Inst,UTD,Dallas,TX,USA.
email_id hmf2014@gmail.com

[I] Abstract :

[Python-Rabin Finger Printing+Python-ImageAI+Python-qrng Library+Z3-Python API +Python-DICOM] in the Context of Processing MRI Scans/cryo-EM Images Using - [IoT/HPC Related Hardware/Software/Firmware/Mathematics] - High Performance Computing Heterogeneous Environments.

key words/index words : Python-Language/ImageAI/qrng-lib/Z3-Theorem Prover/Rabin Finger Printing/DICOM/MRI Scans/cryo-EM Images/IoT/HPC.

[II] Inspiration + Introduction :

https://en.wikipedia.org/wiki/Michael_O._Rabin

<https://rjlipton.wordpress.com/2009/03/01/rabin-flips-a-coin/>

https://en.wikipedia.org/wiki/Rabin_fingerprint

[https://en.wikipedia.org/wiki/Fingerprint_\(computing\)](https://en.wikipedia.org/wiki/Fingerprint_(computing))

rabinfingerprint.org

Fingerprinting Images for Near-Duplicate Detection – Real Python - <https://realpython.com/fingerprinting-images-for-near-duplicate-detection/>

[III] R&D Informatics Framework Using Python Tools Implementation :

[Algorithm I]

input/s ---> [Python-Rabin Finger Printing
+Python-ImageAI+Python-qrng Library
+Z3-Python API +Python-DICOM]---> [**Medical Images/MRI Scans**]
(output/s)

[Algorithm II]

input/s---> [Python-Rabin Finger Printing
+Python-ImageAI+Python-qrng Library
+Z3-Python API +Python-cryoEM Software]---> [**cryo-EM Images**]
(output/s)

[Algorithm III]

input/s---> [Python-Rabin Finger Printing+
Python-ImageAI+Python-qrng Library
+Z3-Python API +Python-DICOM]----> [**IoT/HPC**] ---> [**Medical Images/MRI Scans**]
(output/s)

[Algorithm IV]

input/s---> [Python-Rabin Finger Printing+
Python-ImageAI+Python-qrng Library
+Z3-Python API +Python-cryoEM Software]----> [**IoT/HPC**] ---> [**cryo-EM Images**]
(output/s)

[Figure I – Presents Algorithms [I -IV] – Testing in Progress – Approximate Suggestions Only – Please Check & Satisfy Yourselves – Actual Implementations will certainly vary]

[IV] Related R&D Information on Mathematics+Software Used/Useful :

[a] https://cloudmesh-classes.readthedocs.io/projects/fall-2016/en/latest/python_lesson1.html

[b] <https://realpython.com/fingerprinting-images-for-near-duplicate-detection>

[c] <https://github.com/bastianraschke/pyfingerprint>

[d] <https://pypi.org/project/fingerprint>

[e] <https://stackoverflow.com/.../python-library-for-reading-data-from-fingerprint-devices>

[f] <https://pypi.org/project/fingerprint-app>

[g] <https://subscription.packtpub.com/.../9781785280696/3/cho3lv1sec18/os-fingerprinting>

[h] <https://www.quora.com/Can-Python-read-biometrics-inputs-from-users>

[i] <https://circuitdigest.com/microcontroller-projects/raspberry-pi-fingerprint-sensor...>

[j] <https://www.python.org/downloads>

[k] https://subscription.packtpub.com/book/networking_and_servers/9781785280696/3

[l] <https://github.com/cschwede/python-rabin-fingerprint>

[m] www.sourcecodeonline.com/list?q=rabin_fingerprint

[n] <https://brilliant.org/wiki/rabin-karp-algorithm>

[o] <https://github.com/aitjczize/pyrabin>

[p] <https://github.com/stevegt/librabinpoly>

[q] https://en.m.wikipedia.org/wiki/Rabin_fingerprint

[r] www.cs.cmu.edu/.../15451-f14/www/lectures/lec6/karp-rabin-09-15-14.pdf

[s] <https://www.researchgate.net/publication/2688260>

[t] <https://pypi.org/project/fingerprint>

[u] <https://www.ijser.org/researchpaper/High-Perfomance-Plagiarism...>

[v] <https://github.com/OlafenwaMoses/ImageAI>

[w] <https://pypi.org/project/z3-solver> / <https://ericpony.github.io/z3py-tutorial/guide-examples.htm>

[x] <https://pandas.pydata.org>

[y] <https://pypi.org/project/qrng/> - IBM Stuff - A Quantum Random Number Generator using IBM's Qiskit.

[z] <https://plot.ly/python/visualizing-mri-volume-slices>

[V] Additional Related R&D Information on Mathematics+Software Used/Useful :

- [a1] <https://www.raddq.com/dicom-processing-segmentation-visualization-in-python>
- [a2] <https://school.geekwall.in/p/HylrsP3VPE/intelligent-scanning-using-deep-learning-for-mri>
- [a3] <https://pypi.org/project/mriqc>
- [a4] <https://github.com/jtamir/mri-sim-py>
- [a5] <https://pyscience.wordpress.com/2014/09/08/dicom-in-python-importing-medical-image>.
- [a6] <https://nilearn.github.io>
- [a7] <https://pypi.org/project/scipion-em-eman2>
- [a8] <https://www.ndcn.ox.ac.uk/divisions/fmrib>
- [a9] <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3537914>
- [a10] <https://www.fei.com/cryo-em>
- [a11] <https://www2.mrc-lmb.cam.ac.uk/download/lectures/cryo-em-2017/Cryo-EM...>
- [a12] <https://web.eecs.umich.edu/~fessler/papers/files/talk/o8/siam-plenary.pdf>
- [a13] <https://www.ncbi.nlm.nih.gov/books/NBK232486>
- [a14] <https://iopscience.iop.org/book/978-1-6817-4068-3>
- [a15] <https://pypi.org/project/pydicom>

[VI] Interesting Conclusion/s With Future Perspectives :

We have demonstrated a simple technique using an all Python based solution to perform advanced Software R&D involving MRI Scans/Medical Imaging/cryo-EM Image Processing in the Context of Imaging Mathematics/Rabin Finger Printing(RFP) Algorithms/IoT/QRNG/AI/HPC related Heterogeneous Environments.

[VII] Acknowledgment/s :

Special Thanks to all my Friends+Mentors+Collaborators. Non-Profit R&D.

[VIII] Important References :

[1] An Insight Into Mathematics Behind Rabin Finger Printing to Develop QRNG/ ML-Machine Learning/c/ruby/ruby-LLVM/LLVM-Polly/iot/aot/hpc – High Performance Computing Heterogeneous Systems/environments -

[Source - <http://www.vixra.org/abs/1910.0429>]

[2] http://www.vixra.org/author/nirmal_tej_kumar

[3] http://www.vixra.org/author/d_n_t_kumar

[4] http://www.vixra.org/author/n_t_kumar

[5] <http://www.vixra.org/author/nirmal>

[6] <https://www.semanticscholar.org/author/Nirmal-Tej-Kumar/12354503/suggest>

[7] <https://www.hindawi.com/journals/scn/2018/9081814/>

[THE END]