

PENGOLAHAN MULTI DATA FORMAT RADAR CUACA MENGUNAKAN *WRADLIB* BERBASIS *PYTHON*

MULTI WEATHER RADAR FORMAT DATA PROCESSING USING PYTHON-BASED WRADLIB

Donaldi S Permana^{1*}, Thahir D F Hutapea¹, Alfian S Praja¹, Fatkhuroyan¹, Linda F Muzayanah¹

¹Puslitbang BMKG, Jl. Angkasa I No. 2 Jakarta Pusat 10720 Kemayoran, Jakarta Pusat

*E-mail: donaldi.permana@bmgk.go.id

Naskah masuk:06 September 2016; Naskah diperbaiki:4 Nopember 2016; Naskah diterima:22 Desember 2016

ABSTRAK

Informasi prakiraan cuaca berdasarkan data radar sangat penting bagi BMKG dalam memberikan peringatan dini cuaca ekstrim. Saat ini, BMKG memiliki setidaknya ada tiga format data radar cuaca yang berasal dari tiga produsen radar yakni Gematronik, *Enterprise Electronics Corporation* (EEC) dan Baron yang hanya dapat diolah menggunakan perangkat lunak dari masing-masing produsen radar. Perangkat lunak *wradlib* berbasis *python* dapat mengolah ketiga format data radar tersebut dan menyimpannya dalam format data yang sama. Kelebihan *wradlib-python* lainnya adalah berlisensi sumber terbuka (*open-source*) sehingga dapat di-*install* di berbagai sistem operasi secara gratis, mengurangi ketergantungan terhadap perangkat lunak dari produsen radar, dapat mengolah dan menampilkan multi format data radar cuaca secara masif, menyimpan data radar dalam format NetCDF koordinat kartesian sehingga memudahkan pengolahan data radar lebih lanjut seperti input untuk asimilasi data. Studi ini memfokuskan pengolahan data radar *volumetric* (.vol) luaran produk Gematronik dan Baron serta data radar NetCDF (.nc) luaran produk EEC. *Wradlib-python* dapat mengekstrak secara otomatis data *Plan Position Indicator* (PPI) dan menghitung nilai *Constant Altitude PPI* (CAPPI) dari data reflektifitas radar. Walaupun intensitas reflektifitas citra radar luaran *wradlib-python* relatif lebih tinggi dari luaran perangkat lunak dari produsen radar, luaran tersebut memiliki pola spasial yang relatif sama. Oleh karena itu, *wradlib-python* dapat menjadi salah satu solusi alternatif untuk pengolahan, penyimpanan dan visualisasi data radar cuaca di BMKG.

Kata kunci: multi data format, wradlib, radar cuaca, reflektifitas, koordinat kartesian

ABSTRACT

Weather forecast information based on weather radar is very important for BMKG in providing early warning services for extreme weather. Currently, BMKG has at least three weather radar data format from three radar companies i.e. Gematronik, Enterprise Electronics Corporation (EEC) and Baron which can only be extracted and processed using their original software from each company. The python-based library wradlib can extract and process these three data format and save them into the same data format. Advantages of wradlib-python include having an open-source license so it can be freely installed in multi operating system, reducing dependency on original weather radar software, be able to process and visualize massive multi weather radar data format and be able to save radar data in cartesian coordinate and NetCDF format which make it easier for further data processing such as input for data assimilation. This study only focuses on processing weather radar volumetric data (.vol) from Gematronik and Baron, and NetCDF data (.nc) from EEC. Wradlib-python can automatically extract Plan Position Indicator (PPI) data and calculate Constant Altitude PPI (CAPPI) values from reflectivity data. Although radar images generated by wradlib-python relatively have a greater reflectivity intensity than ones from the original softwares, they have a relatively similar spatial pattern. Therefore, wradlib-python can be one of alternative solutions for extracting, processing and visualization of weather radar data in BMKG.

Keywords: multi format data, wradlib, weather radar, reflectivity, cartesian coordinate

1. Pendahuluan

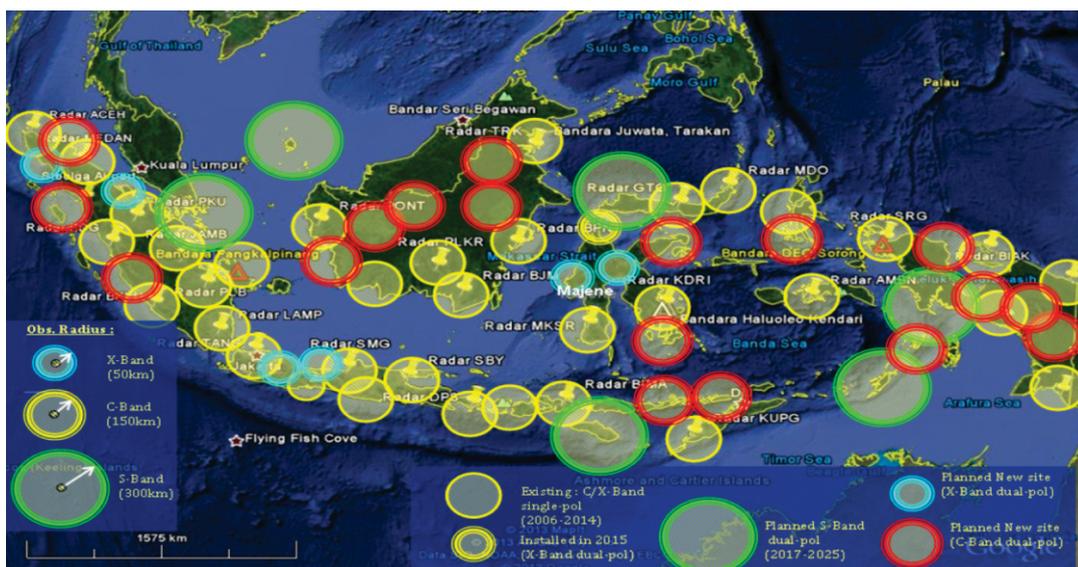
Pada tahun 2016 BMKG telah memiliki 40 perangkat radar cuaca yang tersebar di seluruh wilayah Indonesia dengan komposisi sebagai berikut 20 unit produk EEC, 15 unit produk Gematronik, 4 unit produk Baron dan 1 unit produk Vaisala (sumber : Pusat Meteorologi Publik BMKG). Sedangkan berdasarkan radius jangkauan, radar cuaca BMKG terdiri dari 39 unit tipe *C-band single polarization* dengan maksimum radius jangkauan 150 - 240 km dan 1 unit tipe *X-band dual polarization* dengan maksimum radius jangkauan 50 km. Peta lokasi radar cuaca BMKG ditunjukkan pada Gambar 1. Adapun jumlah unit radar yang sudah terintegrasi sebanyak 37 unit dan hasilnya dapat dilihat pada tautan <http://radar.bmkg.go.id>. Informasi dari radar cuaca sangatlah penting bagi BMKG dalam memberikan pelayanan terkait prakiraan cuaca, khususnya cuaca jangka pendek (*near real time*) terutama untuk pelayanan penerbangan saat *take-off* dan *landing* pesawat di bandara, disamping itu dapat digunakan untuk memberikan pelayanan informasi peringatan dini (*early warning*) cuaca ekstrim seperti akan terjadinya hujan yang sangat lebat, angin puting beliung, angin kencang (*gusty*) dan adanya angin geser (*windshear*) maupun untuk analisa kejadian cuaca ekstrim lainnya.

BMKG memiliki setidaknya tiga produsen radar cuaca yang memiliki perbedaan format data. Tiap format data radar hanya dapat diolah menggunakan perangkat lunak yang berasal dari produsen radar tersebut. Selain itu, data dari beberapa produk radar hanya dapat diolah oleh perangkat lunak yang ter-*install* di lokasi radar

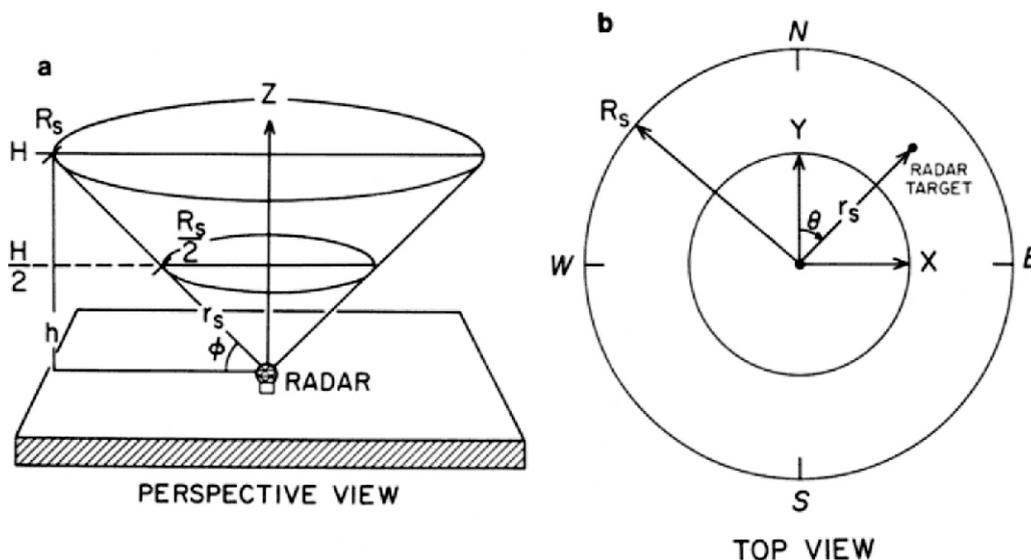
dan stasiun BMKG setempat dikarenakan kendala lisensi. Hal ini menyebabkan kendala pengolahan data radar secara terpusat.

Radar cuaca menghasilkan data secara kontinu setiap ~10 menit dan menyimpan data tersebut pada tempat penyimpanan data lokal di lokasi radar. Tingginya frekuensi pengamatan oleh radar menyebabkan data radar cuaca yang disimpan berukuran cukup besar dan bervariasi tergantung pada jenis produk radar. Kemudian, data tersebut akan ditransfer dan di-*backup* ke stasiun BMKG daerah atau BMKG pusat. Bila proses transfer tidak terjadi dan atau tempat penyimpanan data di lokasi radar sudah penuh, maka data radar yang baru akan menggantikan data yang lama.

Secara umum, data radar cuaca disimpan dalam struktur koordinat polar untuk tiap kemiringan vertikal dari sapuan radar (Gambar 2). Dalam koordinat polar, ada empat komponen utama yang perlu diperhatikan yakni: (1) titik referensi yang merupakan koordinat lokasi radar, (2) sumbu polar, (3) radius yang merupakan jarak antara suatu titik terhadap titik referensi, dan (4) sudut polar (*azimuth*) yang merupakan besar sudut terhadap sumbu polar. Sedangkan, pengolahan data spasial meteorologi biasanya dilakukan dalam struktur koordinat kartesian yang menggunakan koordinat bujur dan lintang. Pengolahan ini biasanya dilakukan dalam proses analisis dan visualisasi data menggunakan perangkat lunak standar yang gratis seperti GrADS, Ferret, NCL, R, Python dan lainnya atau yang berbayar seperti MATLAB dan IDL.



Gambar 1. Peta lokasi radar cuaca BMKG pada tahun 2006-2015 dan rencana instalasi tahun 2015-2024



Gambar 2. (a) Skema observasi data radar dan (b) struktur data radar dalam koordinat polar [1]. h adalah ketinggian di atas permukaan tanah, ϕ adalah sudut kemiringan vertikal sapuan radar, r_s adalah radius target radar dan θ adalah sudut polar (azimuth). Sumber: : <https://www.nssl.noaa.gov/publications/dopplerguide/images/2-1-1.gif>

Oleh karena itu, perlu adanya upaya pengolahan data radar cuaca secara terpusat dengan menggunakan perangkat lunak yang dapat mengolah beberapa format data radar cuaca dan bersifat *open-source*. Selain itu, perangkat lunak juga dapat mengubah data radar dari struktur koordinat polar ke koordinat kartesian dan disimpan dalam format standar, seperti NetCDF. Hal ini nantinya dapat mempermudah proses verifikasi data radar dengan data curah hujan stasiun dan juga proses asimilasi data radar ke dalam model prediksi cuaca numerik.

Jaringan radar terintegrasi yang dikembangkan Pusat Meteorologi Publik BMKG telah menghasilkan produk luaran data radar dengan format yang sama dari beberapa jenis produk data radar cuaca, yaitu format NetCDF. Akan tetapi, struktur data radar yang tersimpan masih dalam koordinat polar sehingga masih berkendala untuk pengolahan lebih lanjut menggunakan perangkat lunak seperti disebutkan diatas. Studi ini bertujuan untuk memberikan alternatif solusi atas kendala yang dihadapi dalam pengolahan dan penyimpanan data radar cuaca di BMKG agar dapat memaksimalkan manfaat data radar cuaca. Kami menggunakan perangkat lunak *open-source wradlib* berbasis *Python* untuk mengolah dan menyimpan data radar cuaca [2-3]. Perangkat lunak ini sudah banyak digunakan dalam pengolahan data radar cuaca dan aplikasinya [4-6]. Sebagai contoh, *wradlib-python* telah digunakan untuk estimasi curah hujan dari data radar dan menunjukkan hasil yang cukup baik dalam estimasi debit aliran sungai dan simulasi kejadian banjir di Filipina [5-6]. Beberapa keuntungan dari penggunaan perangkat lunak ini diantaranya: (1) mengatasi kendala lisensi dan ketergantungan terhadap perangkat lunak yang berasal dari produsen

radar tertentu, (2) mengolah dan mengekstraksi data radar secara masif dengan visualisasi yang dapat disesuaikan, (3) melakukan pemulihan struktur data radar cuaca dari koordinat polar ke dalam koordinat kartesian dan (4) menyimpan luaran data dalam format NetCDF sehingga memudahkan pengolahan data lebih lanjut.

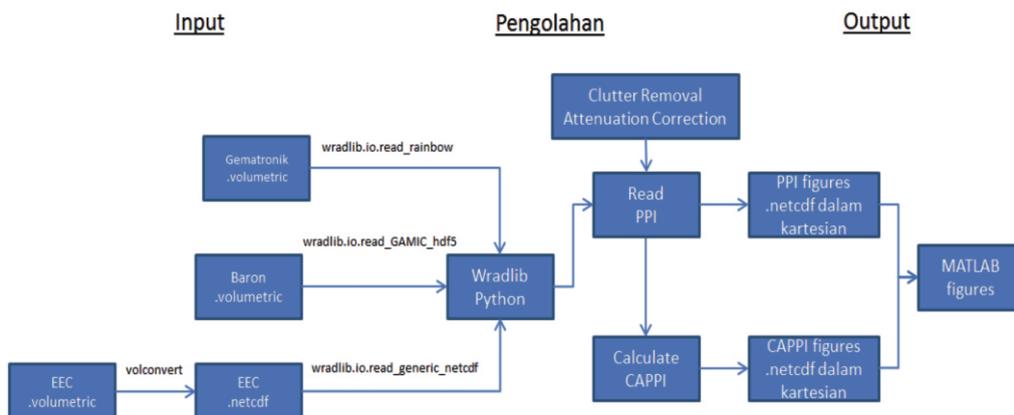
Pada studi ini, tiga format data radar cuaca yang diolah menggunakan *wradlib-python* adalah data *volumetric (.vol)* luaran produk Gematronik dan Baron yang masing-masing biasanya diolah menggunakan perangkat lunak Rainbow® dan FROG-MURAN®, dan data NetCDF (.nc) luaran produk EEC yang biasanya diolah menggunakan perangkat lunak EDGE®. Adapun penyajian studi ini menampilkan proses pengolahan data radar mulai dari ekstraksi data radar sampai dengan penyimpanan dan visualisasi data radar dengan struktur koordinat kartesian dalam format NetCDF. Hasil citra radar luaran *wradlib-python* juga dibandingkan dengan luaran perangkat lunak dari masing-masing produsen radar di beberapa lokasi.

2. Metode Penelitian

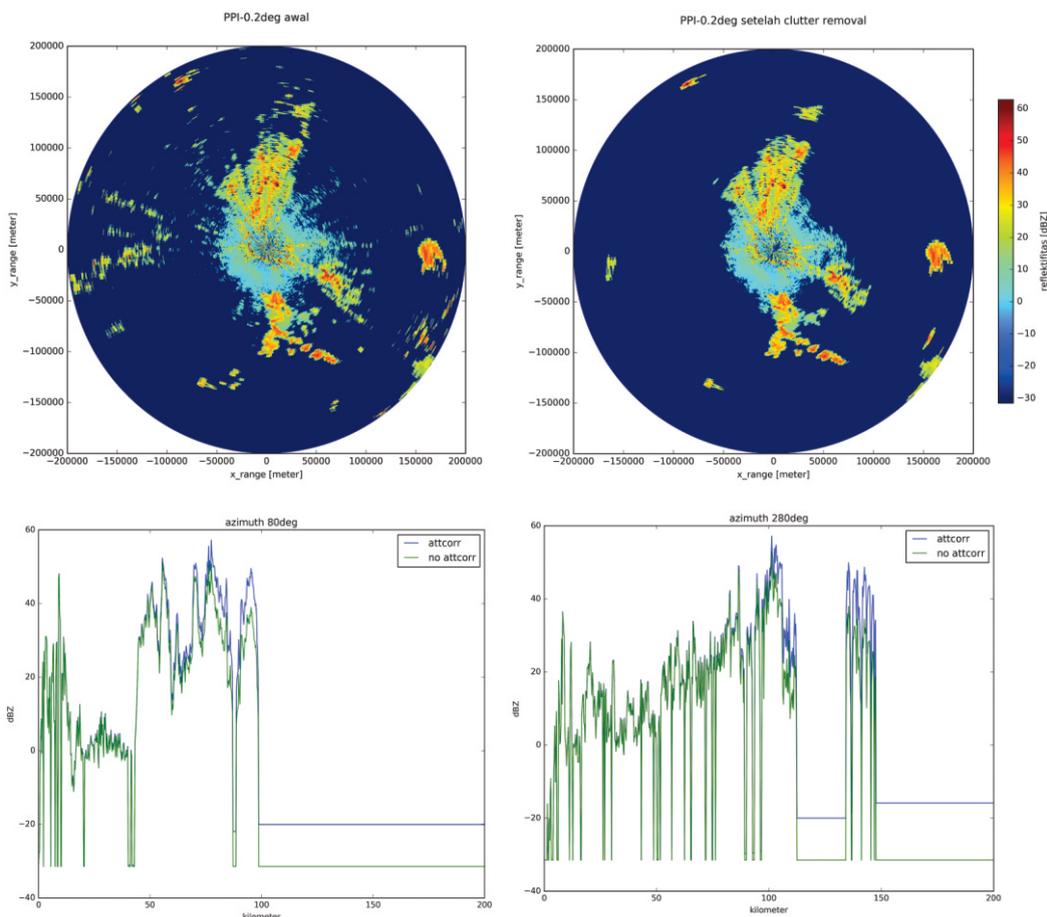
Perangkat lunak *wradlib* adalah salah satu paket (*library*) dari bahasa pemrograman Python yang dikembangkan oleh Universitas Postdam dan Universitas Stuttgart, Jerman [2]. *Wradlib* memiliki fungsi yang penting dalam proses pengolahan data radar cuaca, seperti membaca format data, mendefinisikan koordinat pada peta, mengkonversi reflektivitas ke dalam intensitas curah hujan, mengidentifikasi dan menghapus sinyal kesalahan (*clutter*) dan visualisasi data. *Wradlib* dapat digunakan

untuk mengolah data radar cuaca dari beberapa produk radar dengan format DX dan RADOLAN dari DWD (*German Weather Service*), format *volumetric* dari Gematronik Rainbow, format NetCDF (termasuk luaran dari EEC EDGE), OPERA BUFR, OPERA HDF5/ODIM_H5 dan GAMIC HDF5 [2]. Diagram alur pengolahan data *volumetric* dari tiap format data radar menjadi luaran format NetCDF dalam koordinat kartesian diberikan pada Gambar 3. Tiap data

volumetric radar dapat diekstrak secara langsung oleh *wradlib-python*, kecuali untuk format data radar dari EEC. Data *volumetric* radar EEC harus dikonversi lebih dulu menjadi format NetCDF (.nc) menggunakan paket *volconvert* yang hanya terdapat pada komputer klien yang ter-*install* perangkat lunak EDGE®. Setelah itu, data NetCDF radar EEC baru dapat diekstrak oleh *wradlib-python*.



Gambar 3. Diagram alur pengolahan data *volumetric* dari tiga format data radar.



Gambar 4. Proses *clutter removal* (panel atas) dan *attenuation correction* (panel bawah) pada contoh data radar Palangkaraya tanggal 6 Mei 2016 pukul 15:36 UTC.

Dalam pengolahan data radar, penghapusan sinyal kesalahan (*clutter removal*) yang disebabkan oleh faktor-faktor non-meteorologis seperti adanya objek di permukaan bumi (pegunungan, bukit, gedung tinggi) atau objek di udara (pesawat udara, burung, dan sebagainya) telah dilakukan oleh *clutter filter* yang dikembangkan oleh Gabella dan Notarpietro (2002) [7]. Sedangkan, kesalahan atenuasi (*attenuation correction*) yang biasa disebabkan oleh *radome* (penutup radar) yang basah dan kondisi hujan lebat juga telah dikoreksi menggunakan metode yang dikembangkan oleh Kraemer dan Verworn (2009) [8]. Contoh proses *clutter removal* dan *attenuation correction* diberikan pada Gambar 4. Pada Gambar 4 (panel atas) ditunjukkan bahwa *clutter removal* dapat menghapus *noise* yang terekam dalam radar sedangkan penguatan sinyal radar yang berjarak jauh dari pusat radar dilakukan pada tahap *attenuation correction* (panel bawah).

Beberapa skrip pemrograman *python* telah dibuat menggunakan aplikasi gratis *Scientific Python Development Environment* (Spyder) untuk mengekstrak, mengolah input data *volumetric* (.vol) dari radar Gematronik dan Baron maupun data NetCDF (.nc) dari radar EEC dan menyimpan luaran data radar dalam koordinat kartesian dengan format NetCDF. Tiap skrip dirancang untuk mengekstrak dan menampilkan data reflektifitas tiap sudut kemiringan vertikal radar (PPI, Gambar 2) secara otomatis dan untuk menghitung nilai CAPPI, yaitu tampilan data reflektifitas radar secara horisontal pada ketinggian tetap tertentu, dan juga nilai maksimal CAPPI (CAPPI-CMAX) pada kolom ketinggian. Adapun spesifikasi perhitungan CAPPI yang telah dirancang adalah (1) memiliki resolusi horisontal 0.5 km/pixel, (2) luaran nilai CAPPI untuk ketinggian 0.5 sampai dengan 5 km dengan resolusi vertikal 0.5 km, dan (3) luaran nilai CAPPI-CMAX pada kolom ketinggian 0.5 - 5 km.

Pengolahan data radar menggunakan komputer dengan spesifikasi processor Intel(R) Core(TM) i7, dibutuhkan waktu rata-rata sekitar 1 - 3 menit untuk mengekstrak PPI/CAPPI dari data reflektifitas radar dan menyimpannya kedalam format NetCDF4 untuk data radar tiap 10 menit. Luaran file NetCDF PPI/CAPPI data reflektifitas yang terbentuk berukuran sekitar 4 - 7 MB untuk data radar per 10 menit tergantung kepada radius jangkauan radar, jumlah PPI dan jumlah ketinggian CAPPI yang disimpan.

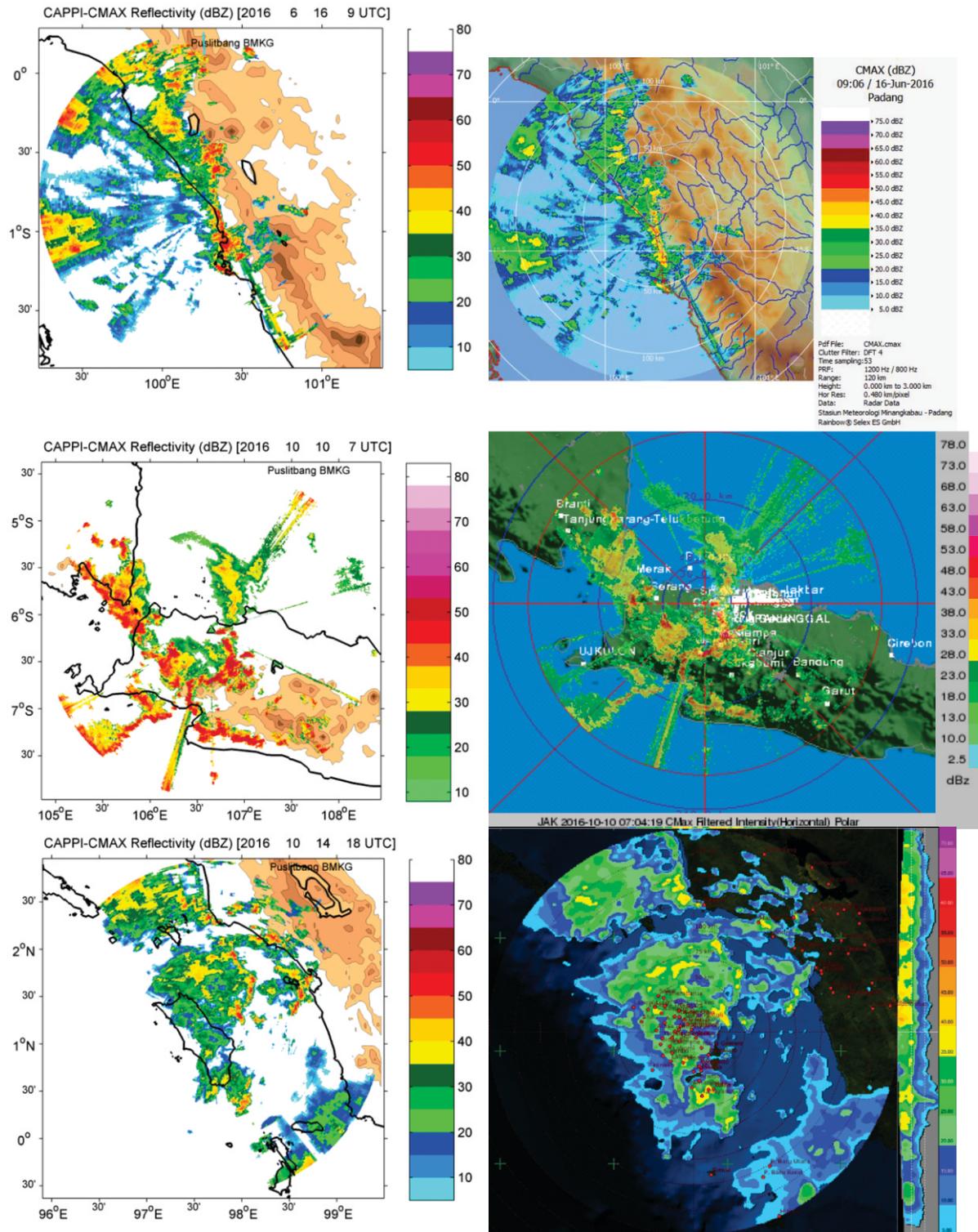
3. Hasil dan Pembahasan

Bagian ini membandingkan dan membahas contoh luaran CAPPI-CMAX data reflektifitas radar antara perangkat lunak *wradlib-python* dan perangkat lunak

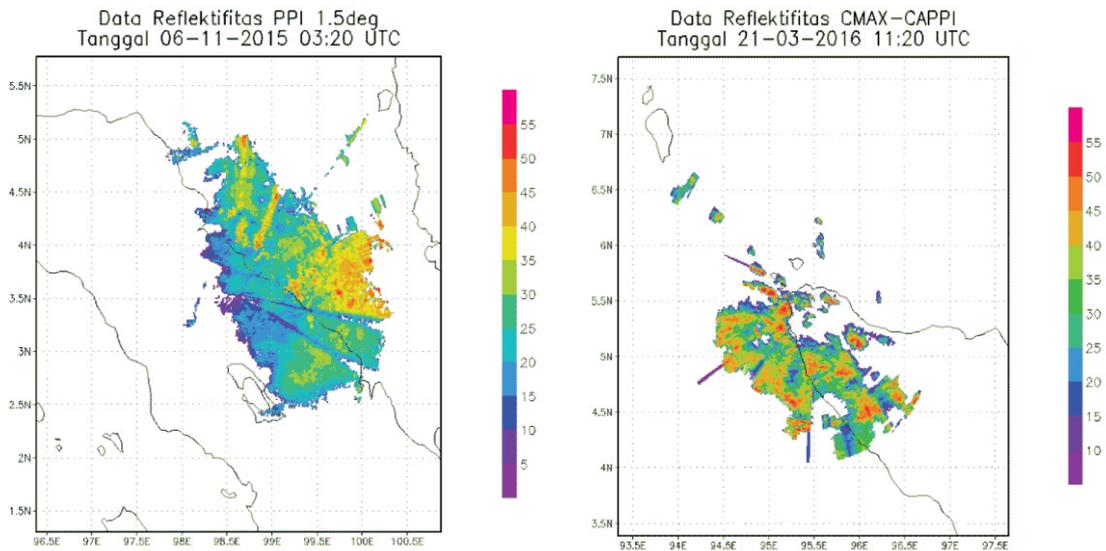
dari produsen radar. Pada studi ini, perbandingan citra radar antara luaran *wradlib-python* dengan luaran perangkat lunak Rainbow® dari Gematronik, EDGE® dari EEC dan FROG-MURAN® dari Baron hanya dilakukan secara spasial dan intensitas, dan belum secara temporal. Perbedaan yang tampak antara luaran tersebut dijelaskan dengan memberikan kemungkinan faktor-faktor yang mempengaruhi luaran citra radar. Selain itu, hasil penyimpanan data reflektifitas radar dalam koordinat kartesian dan format NetCDF divisualisasikan menggunakan perangkat lunak GrADS.

Pada studi ini, analisa temporal (*time series*) perbandingan citra radar luaran *wradlib-python* dan luaran perangkat lunak dari produsen radar belum dapat dilakukan karena kendala akses data digital dari luaran perangkat lunak produsen radar yang hanya ter-*install* pada lokasi radar. Gambar 5 membandingkan luaran tersebut dan menunjukkan hasil dengan pola spasial yang relatif sama. Akan tetapi, secara umum intensitas reflektifitas radar luaran *wradlib-python* lebih tinggi (*overestimate*) dari luaran perangkat lunak dari produsen radar. Adapun perbedaan pola spasial dan intensitas reflektifitas yang tampak dapat disebabkan oleh beberapa faktor diantaranya perbedaan resolusi horisontal antara luaran kedua perangkat lunak, perbedaan metode interpolasi data spasial, perbedaan metode *clutter removal* dan *attenuation correction*, dan lain sebagainya.

Sebagai contoh, Gambar 5 (panel atas kanan) menunjukkan luaran CMAX Rainbow® dari Gematronik untuk kasus kejadian hujan ekstrim yang menyebabkan banjir di kota Padang tanggal 16 Juni 2016 [9-10]. Secara spasial, luaran CAPPI-CMAX *wradlib-python* (panel atas kiri) menyerupai luaran Rainbow® yang menggambarkan adanya *squall line* pada arah utara-selatan yang melintasi kota Padang (pusat radar) dan menyebabkan hujan ekstrim dan kejadian banjir. Namun, pada intensitas reflektifitas dari luaran *wradlib-python* lebih tinggi dibandingkan dengan luaran Rainbow®, terutama pada area dengan nilai reflektifitas tinggi. Untuk area dengan nilai reflektifitas rendah, kedua luaran tersebut mempunyai intensitas yang relatif sama. Perbedaan tersebut dapat disebabkan oleh perhitungan CAPPI-CMAX pada *wradlib-python* menggunakan ketinggian 0 - 5 km sedangkan pada Rainbow® menggunakan ketinggian 0 - 3 km. Selain itu, perbedaan metode *clutter removal* antara *wradlib-python* (Gabella dan Notarpietro (2002) [7]) dan Rainbow® (DFT 4) juga memungkinkan terjadinya perbedaan pola spasial dan intensitas reflektifitas.



Gambar 5. (Panel atas) perbandingan citra data reflektifitas antara luaran *wradlib-python* (kiri) dan luaran Rainbow® (kanan) [9-10] untuk kasus data radar Padang tanggal 16 Juni 2016 pukul 09:06 UTC. Luarannya Rainbow® dipersiapkan oleh Stasiun Meteorologi Minangkabau, Padang. (Panel tengah) perbandingan citra data reflektifitas antara luaran *wradlib-python* (kiri) dan luaran EDGE® (kanan) untuk kasus data radar Cengkareng tanggal 10 Oktober 2016 pukul 00:00 UTC. Luarannya EDGE® dipersiapkan oleh Stasiun Meteorologi Soekarno-Hatta, Jakarta. (Panel bawah) perbandingan citra data reflektifitas antara luaran *wradlib-python* (kiri) dan luaran FROG-MURAN® (kanan) untuk kasus data radar Nias tanggal 14 Oktober 2016 pukul 18:00 UTC. Luarannya FROG-MURAN® dipersiapkan oleh Stasiun Meteorologi Binaka, Gunungsitoli, Nias.



Gambar 6. Plot data reflektifitas radar PPI (Medan, kiri) dan CMAX-CAPPI (Banda Aceh, kanan) dengan format NetCDF4 menggunakan GrADS.

Metode interpolasi data *nearest-grid* yang digunakan *wradlib-python* pada studi ini juga dapat menyebabkan perbedaan pola spasial dan intensitas reflektifitas. Adanya perbedaan metode *attenuation correction* antara kedua perangkat lunak tersebut juga memberikan pengaruh terhadap intensitas reflektifitas, terutama pada area yang berlokasi jauh dari pusat radar.

Perbandingan luaran *wradlib-python* dengan luaran EDGE® dan luaran FROG-MURAN® juga ditunjukkan masing-masing pada Gambar 5 panel tengah untuk radar Jakarta dan panel bawah untuk radar Nias. Secara spasial, luaran radar tersebut menunjukkan pola yang relatif sama. Akan tetapi, secara umum intensitas reflektifitas luaran *wradlib-python* lebih tinggi dari luaran EDGE® dan FROG-MURAN®. Khusus untuk data radar EEC (panel tengah), selisih nilai reflektifitas radar antara luaran *wradlib-python* dan luaran EDGE® dapat mencapai 10 dBZ. Hal ini perlu dikaji lagi untuk menentukan penyebab terjadinya perbedaan signifikan tersebut. Faktor yang paling memungkinkan adalah perbedaan metode *clutter removal* dan *attenuation correction* antara kedua perangkat lunak tersebut. Selanjutnya, perbedaan luaran *wradlib-python* untuk data radar BARON dengan luaran FROG-MURAN® secara umum hampir sama dengan perbedaan luaran Rainbow® untuk data Gematronik, yakni memiliki pola spasial yang relatif sama dan intensitas yang relatif lebih tinggi. Hal ini juga mungkin disebabkan oleh faktor-faktor yang telah dijelaskan sebelumnya.

Pada Gambar 5 masih terlihat adanya *noise* yang terekam pada luaran *wradlib-python* setelah proses

clutter removal. Hal ini mungkin disebabkan karena adanya interferensi frekuensi radar dengan frekuensi jaringan komunikasi di sekitar lokasi radar yang muncul dan menguat pada waktu-waktu tertentu sehingga *clutter removal* belum dapat menghilangkannya. Kemungkinan lainnya disebabkan oleh penggunaan metode *clutter removal* yang belum optimal. Hal ini mungkin dapat diatasi dengan melakukan pemilihan metode *clutter removal* yang paling tepat dan optimal. Di samping itu, *noise* juga masih terekam pada luaran Rainbow® dan EDGE® menunjukkan bahwa interferensi frekuensi jaringan komunikasi disekitar lokasi radar cukup kuat sehingga tidak terhapus oleh proses *clutter filter* pada perangkat lunak dari produsen radar.

Seperti ditunjukkan pada diagram alur (Gambar 3), *wradlib-python* menghasilkan luaran data radar dengan format NetCDF dalam koordinat kartesian untuk PPI dan CAPPI. Hal ini juga merupakan kelebihan *wradlib-python* dibandingkan dengan Rainbow®, EDGE® dan FROG-MURAN®. Penyimpanan data dalam koordinat kartesian memudahkan pengguna dalam pengolahan data radar lebih lanjut seperti proses validasi dengan data observasi curah hujan, proses asimilasi data dengan model prakiraan cuaca numerik dan pembentukan mosaik integrasi radar. Salah satu contoh penggunaan data radar dengan format NetCDF dalam koordinat kartesian adalah dapat divisualisasi menggunakan perangkat lunak *Grid Analysis and Data System* (GrADS). Sebagai contoh, Gambar 6 menunjukkan tampilan luaran data reflektifitas radar menggunakan GrADS untuk data radar kota Medan (EEC) dan Banda Aceh (Gematronik).

4. Kesimpulan

Secara umum, *wradlib-python* dapat mengatasi permasalahan BMKG yang memiliki multi format data radar cuaca, kendala lisensi, pengolahan data radar terpusat dan ketergantungan terhadap perangkat lunak yang berasal dari produsen radar tertentu. *Wradlib-python* dapat mengekstrak tiap sudut kemiringan vertikal radar (PPI) secara otomatis, mengolah, dan menampilkan data radar secara masif, dan menyimpan luaran data radar dalam koordinat kartesian dan format NetCDF sehingga memudahkan untuk pengolahan data radar lebih lanjut.

Perbandingan citra radar antara *wradlib-python* dan perangkat lunak dari produsen radar menunjukkan luaran citra reflektifitas radar dengan pola spasial yang relatif sama. Akan tetapi, intensitas reflektifitas luaran *wradlib-python* relatif lebih tinggi (*overestimate*) dibandingkan dengan luaran perangkat lunak dari produsen radar. Hal ini dapat disebabkan oleh beberapa faktor diantaranya perbedaan resolusi horisontal citra luaran, metode interpolasi data spasial, metode *clutter removal* dan *attenuation correction*. Oleh karena itu, *wradlib-python* dapat dijadikan sebagai salah satu solusi alternatif untuk pengolahan, penyimpanan dan visualisasi data radar cuaca di BMKG.

Saran. Penulis merekomendasikan penggunaan *wradlib-python* sebagai solusi alternatif untuk pengolahan dan visualisasi data radar cuaca di Pusat Meteorologi Publik dan untuk penyimpanan data radar cuaca di Pusat Database BMKG.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Kepala Pusat Penelitian dan Pengembangan dan Kepala Bidang Penelitian dan Pengembangan Meteorologi serta rekan-rekan staf Bidang Penelitian dan Pengembangan Meteorologi BMKG atas dukungan, saran, dana dan sarana prasarana dalam pelaksanaan penelitian dan publikasi artikel ini.

Daftar Pustaka

[1] Brown, R. A., and Wood, V. T.,: A guide for interpreting Doppler velocity patten: Northern Hemisphere Edition, NOAA/National Severe Storms Laboratory Norman, Oklahoma, Second Edition, <https://www.nssl.noaa.gov/publications/dopplerguide/chapter2.php>, 2007

[2] Pfaff, T., Heistermann, M., and Jacobi, S.: wradlib - An open source library for weather radar data processing, In Proceedings of ERAD - The Seventh European Conference on Radar in Meteorology and Hydrology, 2012

[3] Heistermann, M., Jacobi, S., and Pfaff, T.: Technical Note: An open source library for processing weather radar data (wradlib), *Hydrol. Earth Syst. Sci.*, 17, 863-871, doi:10.5194/hess-17-863-2013, 2013.

[4] Heistermann, M., Crisologo, I., Abon, C. C., Racoma, B. A., Jacobi, S., Servando, N. T., David, C. P. C., and Bronstert, A.: Brief communication" Using the new Philippine radar network to reconstruct the Habagat of August 2012 monsoon event around Metropolitan Manila". *Natural Hazards and Earth System Sciences*, 13(3), 653-657, 2013

[5] Abon, C., Crisologo, I., Kneis, D., Bronstert, A., and Heistermann, M.: Comparison between C-band and S-band radar rainfall for hydrological simulations in Marikina River Basin, Philippines, In Proceedings of ERAD - The Eight European Conference on Radar in Meteorology and Hydrology, 2014

[6] Abon, C. C., Kneis, D., Crisologo, I., Bronstert, A., David, C. P. C., & Heistermann, M.: Evaluating the potential of radar-based rainfall estimates for streamflow and flood simulations in the Philippines. *Geomatics, Natural Hazards and Risk*, 7(4), 1390-1405, 2016

[7] Gabella, M. and Notarpietro, R.: Ground clutter characterization and elimination in mountainous terrain. In Proceedings of ERAD 2002, Delft, The Netherlands, 305-311, available at: <http://www.copernicus.org/erad/online/erad-305.pdf>, 2002

[8] Kraemer, S. and Verworn, H. R.: Improved radar data processing algorithms for quantitative rainfall estimation in real time, *Water Sci. Technol.*, 60, 175-184, 2009

[9] Wulandari, E, S, P., and Nugraha, Y.: "Analisis cuaca terkait kejadian hujan ekstrim di Sumatera Barat mengakibatkan banjir dan genangan air di kota Padang tanggal 16 Juni 2016." *I n t e r n e t* : http://eoffice.bmkg.go.id/Dokumen/Artikel/Artikel_20160628154826_eavde8_Analisis-Cuaca-Terkait-Kejadian-Hujan-Ekstrim-di-Sumatera-Barat-16-Juni-2016.pdf, diakses 24 Juli 2016

[10] Muzayanah, L. F., Permana, D. S., Praja, A. S., and Wulandari, E, S, P.: Extreme rainfall analysis using radar-based rainfall estimates, ground observation and model simulation in West Sumatra (case study: Padang floods on June 16, 2016), In Proceedings of International Symposium on the 15th Anniversary of the Equatorial Atmosphere Radar (EAR), Jakarta, 4 August 2016.