

Spatio-Temporal Clustering Hotspot di Sumatera Selatan Tahun 2002-2003 Menggunakan Algoritme ST-DBSCAN dan Bahasa Pemrograman R

Spatio-Temporal Clustering Hotspot in South Sumatera From 2002-2003 Using ST-DBSCAN Algorithm and R Programming Language

NADINA ADELIA INDRAWAN, HARI AGUNG ADRIANTO*

Abstrak

Kebakaran hutan merupakan permasalahan yang cukup serius di Indonesia. Salah satu indikator kemungkinan terjadinya kebakaran hutan dapat diketahui dengan kemunculan *hotspot*. *Dataset hotspot* merupakan data spasial yang berukuran besar karena dicatat setiap waktu. *Spatial temporal-Density based clustering algorithm*(ST-DBSCAN) merupakan salah satu algoritme yang mampu mengolah data spasial dan temporal. Penelitian ini mengimplementasikan algoritme ST-DBSCAN menggunakan bahasa pemrograman R. R adalah perangkat lunak komputasi statistik dan grafis. Bahasa R digunakan di kalangan ahli statistik dan data *mining* untuk mengembangkan perangkat lunak statistik dan analisis data. *Clustering* dilakukan terhadap *dataset hotspot* di Sumatera Selatan pada periode tahun 2002-2003. Dengan menggunakan parameter jarak spasial ($Eps_1 = 0.2$), parameter jarak temporal ($Eps_2 = 7$) dan minimum anggota *cluster* ($MinPts = 7$) didapatkan hasil 41 *cluster* dengan 712 *noise*.

Kata Kunci: kebakaran hutan, R, *spatio-temporal*, ST-DBSCAN, titik panas.

Abstract

Forest fire is a serious problem in Indonesia. One indicator of forest fires possibility can be seen through the appearance of hotspots. Hotspot dataset is large spatial data because its recorded each time. ST-DBSCAN is one of the algorithm which has the capability to process spatial and temporal data. This study is implementing Spatial temporal-Density based clustering algorithm(ST-DBSCAN) with R language programming. R is software environment for statistical computing and graphics. The R language is widely used among statisticians and data miners for developing statistical software. Clustering is performed on hotspots dataset in South Sumatra in the period 2002-2003. By using the spatial distance parameter ($Eps_1 = 0.2$), the temporal distance parameters ($Eps_2 = 7$) and a minimum cluster members ($MinPts = 7$) showed 41 clusters with 712 noise.

Keywords: forest fire, hotspot, R, spatio-temporal, ST-DBSCAN.

PENDAHULUAN

Kebakaran hutan di Indonesia merupakan masalah yang terus berulang dan termasuk permasalahan yang serius. Menurut Adinugroho (2005) salah satu indikator kemungkinan terjadinya kebakaran hutan ialah *hotspot*. Dalam hal ini terdapat kemungkinan bahwa persebaran titik api menggerombol dalam ruang secara alami tidak tersebar acak sehingga data tersebut dapat dianalisis menggunakan teknik *clustering*.

Ada beberapa macam teknik *clustering* diantaranya *k-means*, *hierachical cluster*, DBSCAN dan ST-DBSCAN. Di antara semua metode *clustering*, algoritme DBSCAN merupakan salah satu metode yang paling baik untuk menemukan *cluster* dari *database* spasial yang besar (Gaonkar & Sawant, 2013). ST-DBSCAN merupakan perluasan algoritme

DBSCAN, jika DBSCAN hanya mengolah data *spatial* maka ST-DBSCAN mampu mengolah data temporal dengan menggunakan nilai Eps_1 (parameter jarak pada aspek spasial) dan nilai Eps_2 (parameter jarak pada aspek temporal). ST-DBSCAN memiliki kelebihan lebih fleksibel dengan ukuran data yang besar karena merupakan teknik *clustering* yang berbasis kepadatan (Birant & Kut, 2007).

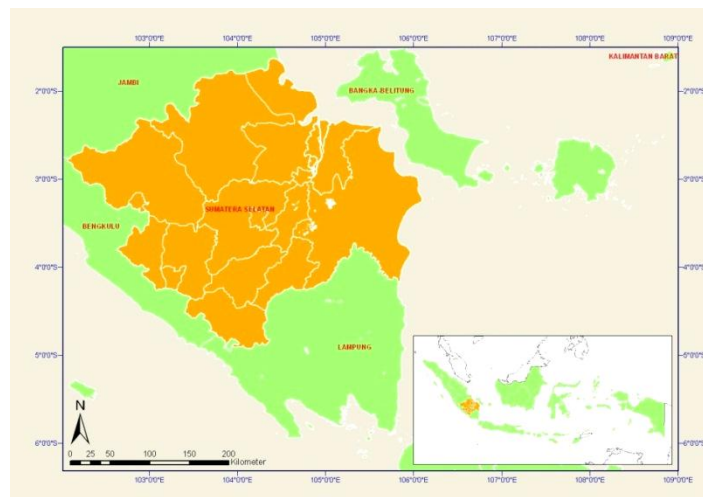
R adalah perangkat lunak komputasi statistik dan grafis. R sangat efektif dalam hal memvisualisasikan *cluster* dan dalam pengelolaan data serta fasilitas penyimpanan. Selain itu R dapat dikembangkan sesuai kebutuhan dengan menambah fitur-fitur tambahan dalam bentuk paket ke dalam *software* R yang bersifat *open source* (Venables & Smith, 2009).

Pada penelitian diimplementasikan algoritme ST-DBSCAN menggunakan bahasa pemrograman R. Data yang dipakai ialah *dataset hotspot* di Sumatera Selatan tahun 2002-2003. Diharapkan implementasi ST-DBSCAN dapat menghasilkan pola penyebaran *hotspot* sehingga kebakaran hutan di Indonesia dapat dikendalikan.

METODE

Data Penelitian

Wilayah penelitian melingkupi Provinsi Sumatera Selatan yang terletak antara $5^{\circ}10'$ sampai $1^{\circ}20'$ Lintang Selatan dan $101^{\circ}40'$ sampai $106^{\circ}30'$ Bujur Timur (Gambar 3). Penelitian ini menggunakan *dataset hotspot* tahun 2002-2003 dengan 1° mewakili jarak sekitar 110 km.

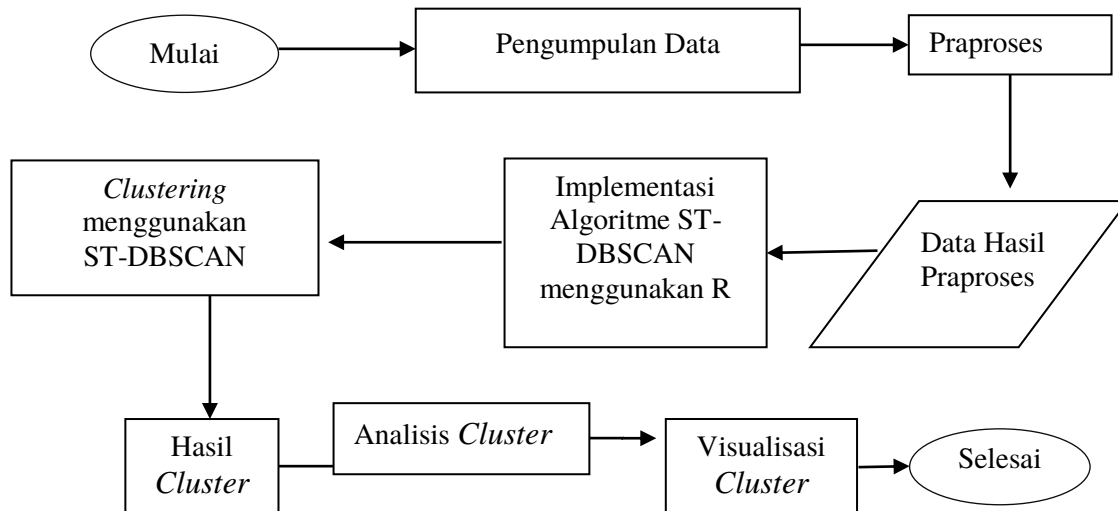


Gambar 1 Peta Provinsi Sumatera Selatan (Purwanto, 2012)

Data penelitian diperoleh dari Fire Information For Resource Management System (FIRMS) yang merupakan data *hotspot* MODIS yang disediakan oleh National Aeronautics and Space Administration (NASA). Variabel yang terdapat pada *dataset* meliputi titik spasial (*latitude* dan *longitude*), tingkat kecerahan pada piksel *hotspot* (Kelvin), ukuran piksel sebenarnya (*scan* dan *track*), tanggal dan waktu pengambilan data, satelit (Aqua dan Terra), *confidence* yang menunjukkan ukuran kualitas piksel pada setiap *hotspot* (skala 0-100 %), versi pengambilan data (*collection* – angka sebelum desimal dan *source* – angka sesudah desimal yang mengindikasikan sumber data), dan FRP (menunjukkan piksel yang terintegrasi radiasi api dalam satuan MegaWatts).

Tahapan Penelitian

Tahapan penelitian yang dilakukan pada penelitian ini ditunjukkan pada Gambar 4.



Gambar 2 Diagram alir penelitian

Praproses Data

Tahap praproses data terdiri atas tahapan transformasi data dan reduksi data. Pada tahap transformasi dilakukan perubahan tipe format data dengan merubah tipe format *date* menjadi tipe format *number*. Selain itu dilakukan tahapan reduksi data dengan melakukan pengurangan data dari 11 variabel menjadi 3 variabel yaitu *longitude*, *latitude* dan tanggal pemantauan *hotspot*.

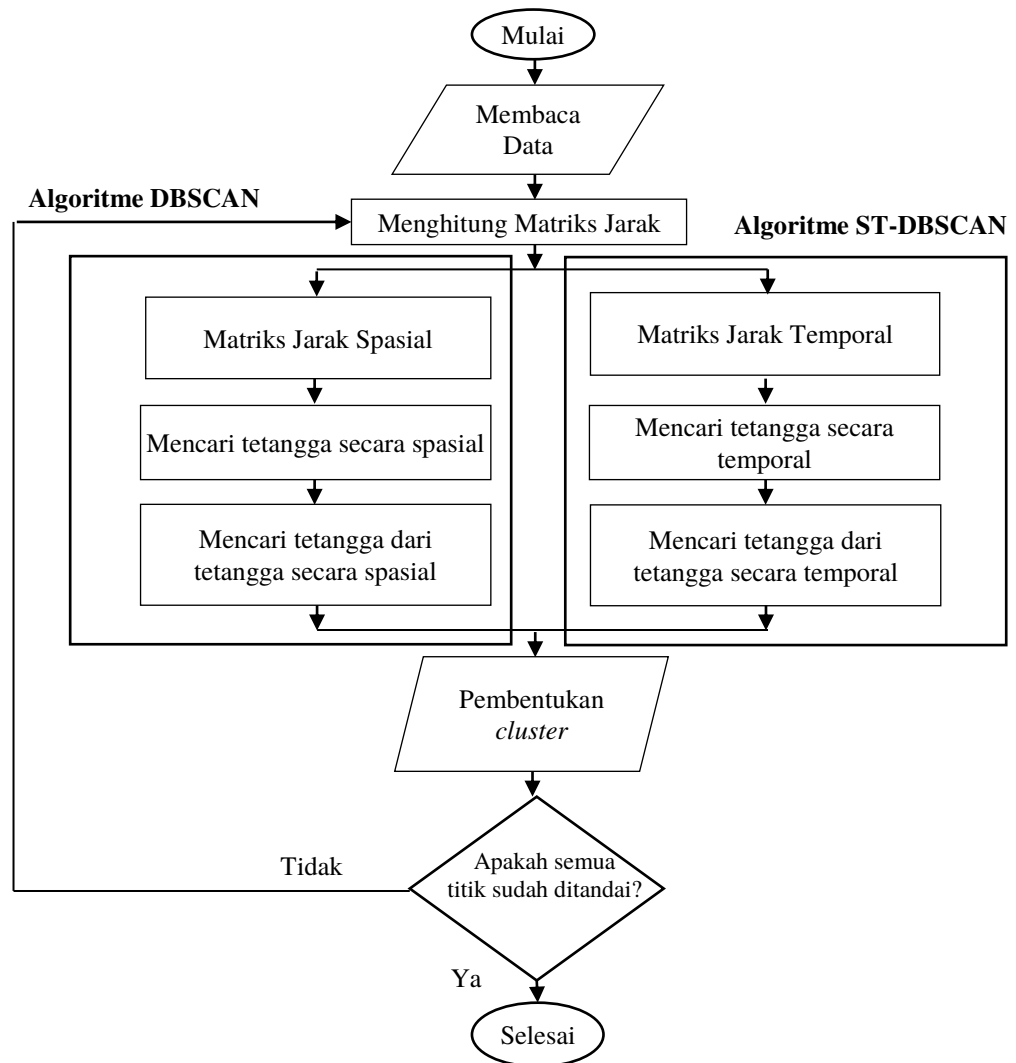
Clustering Menggunakan ST-DBSCAN

Algoritme ST-DBSCAN merupakan pengembangan dari algoritme DBSCAN. Algoritme ST-DBSCAN membutuhkan empat parameter, yaitu Eps_1 (parameter yang menyatakan jarak antar 2 objek di permukaan bumi), Eps_2 (parameter yang menyatakan jarak antar kejadian waktu), $MinPts$ (jumlah anggota suatu *cluster*), dan $\Delta \in$ (Brian & Kut, 2007). Pada penelitian ini hanya tiga parameter yang digunakan yaitu Eps_1 , Eps_2 dan $MinPts$.

Penentuan nilai parameter merujuk pada penelitian Purwanto (2012). Penentuan nilai ketiga parameter tersebut dilakukan agar mendapatkan total *cluster* dan *noise* dalam jumlah yang sedikit.

Implementasi Algoritme ST-DBSCAN Menggunakan R

Pada tahapan ini dilakukan modifikasi fungsi *dbscan* yang terdapat pada *package fpc*. Fungsi *dbscan* pada *package fpc* hanya menggunakan satu epsilon kemudian dimodifikasi menjadi *st-dbscan* dengan dua epsilon. Alur algoritme DBSCAN dan ST-DBSCAN dapat dilihat pada Gambar 3.



Gambar 3 Diagram alir algoritme

Analisis Cluster

Pada tahapan ini akan dianalisis apakah *cluster* tersebut memiliki pola *stationary*, *reappearing* (*regular* dan *irregular*), *occasional*, ataukah *track* (Pöelitz & Andrienko, 2010).

Lingkupan Pengembangan

Pembuatan model *clustering* dengan menggunakan algoritme ST-DBSCAN ini menggunakan beberapa perangkat keras dan perangkat lunak dengan spesifikasi sebagai berikut:

Perangkat Keras

Processor: Intel(R) Core(TM) i3 CPU M 2330 @ 2.2GHz

Memory: Sodim DDR3 2 GB RAM

Monitor

Perangkat Lunak

Sistem operasi Windows 7 Enterprise 64-bit

Bahasa Pemrograman R-3.0.2 version of R64

Microsoft Excel 2013

Notepad++

HASIL DAN PEMBAHASAN

Praproses Data

1. Perubahan format pada kolom *date*

Data tanggal akan di ubah ke dalam format *number* menggunakan Microsoft Excel dimana hari pertama dimulai dari 1 Januari 1900. Pengubahan format ini dilakukan agar ketika terjadi penambahan data sebelum tanggal pertama data yang digunakan saat ini tidak menyebabkan variabel tanggal bernilai -1 jika penanggalan dimulai dari 1 Januari 2008.

Tabel 3 Transformasi data mengubah format *date* menjadi *number*

Tanggal (mm/dd/yyyy)	Tanggal (<i>number</i>)
11/05/2002	37387
22/05/2002	37398
29/05/2002	37405
03/06/2002	37410

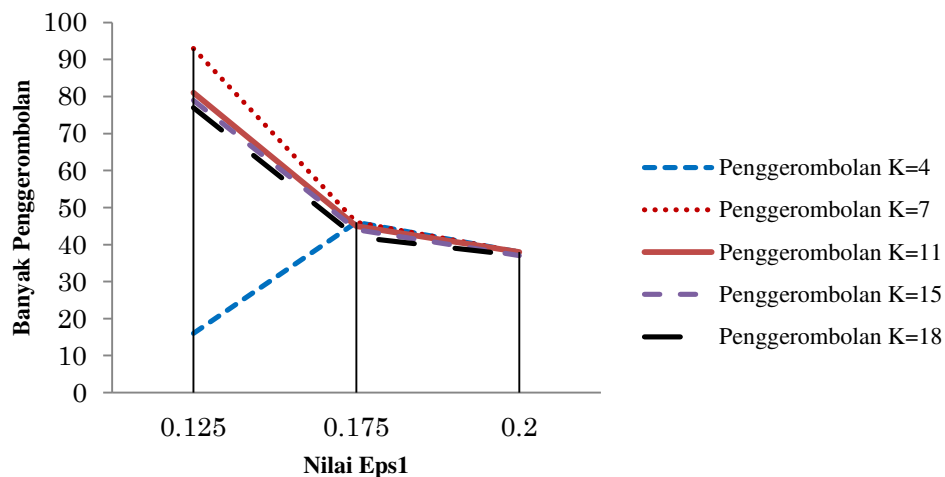
2. Reduksi data

Penelitian ini hanya memerlukan aspek spasial dan aspek temporal untuk proses *mining* sehingga reduksi data dilakukan dengan memilih 3 atribut dari 11 atribut. Atribut yang terdapat dalam *dataset* yaitu: *longitude*, *latitude*, *acq_date*, *acq_time*, *brightness*, *confidence*, *satelite*, *bright_t31*, *frp*, *scan* dan *track*. Atribut yang dipilih yaitu *longitude*, *latitude* yang merupakan aspek spasial dan tanggal pemantauan *hotspot* sebagai aspek temporalnya.

3. Perhitungan nilai parameter

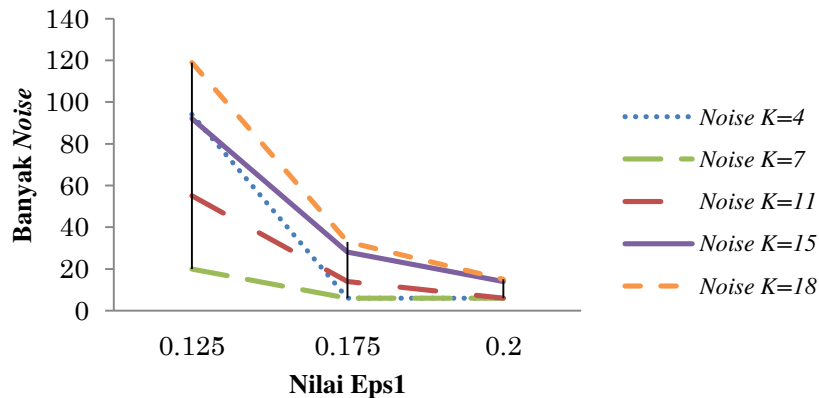
Pada penelitian Purwanto (2012) penentuan nilai Eps_1 dan $MinPts$ dilakukan dengan perhitungan *k-dist* dengan nilai k yang bervariasi. Penggerombolan dapat diidentifikasi jika k tetangga terdekat memiliki jarak yang sama dengan titik yang lain atau dapat dikatakan bahwa titik dalam suatu penggerombolan secara kasar memiliki jarak yang sama.

Optimalisasi nilai Eps_1 dan $MinPts$ (k) dilakukan dengan cara melakukan pergeseran nilai Eps_1 (Gambar 5). Pergeseran nilai Eps_1 dilakukan pada nilai $k=4$, $k=7$, $k=15$ dan $k=18$ dengan nilai Eps_1 0.125 sampai 0.2. Dari pergeseran nilai Eps_1 didapatkan nilai $Eps_1 = 0.2$ dengan total hasil *cluster* paling sedikit sehingga pada penelitian ini digunakan $Eps_1 = 0.2$ yang mewakili jarak sekitar 22 km.



Gambar 4 Grafik pergeseran penggerombolan (Purwanto 2012)

Gambar 6 menunjukkan jumlah titik yang tidak terdapat di *cluster* manapun (*noise*) pada $k=7$ lebih stabil seiring penambahan nilai Eps_1 . Berdasarkan nilai $Eps_1=0.2$ diperoleh $k = 4$, $k = 7$ dan $k = 11$ memiliki *noise* paling sedikit sehingga pada penelitian ini digunakan $minpts = 7$ dan $Eps_1=0.2$.

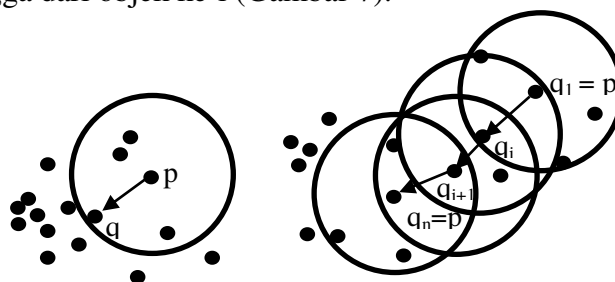


Gambar 5 Hubungan titik noise dengan Eps_1 (Purwanto, 2012)

Pada ST-DBSCAN selain parameter Eps_1 dan $MinPts$ juga digunakan parameter Eps_2 . Pada penelitian ini nilai Eps_2 yang dipakai adalah 7 meskipun Purwanto (2012) menggunakan nilai 30 sebagai Eps_2 . $Eps_2 = 7$ bertujuan agar perubahan cuaca yang terjadi tidak jauh berbeda dan untuk mendapatkan pola dalam rentang waktu yang cenderung lebih kecil.

Implementasi Algoritme ST-DBSCAN Menggunakan R

Tahap ini dilakukan dengan mengimplementasikan algoritme ST-DBSCAN ke dalam bahasa pemrograman R (Brian & Kut, 2007). Algoritme dimulai dengan perhitungan matriks jarak *euclidean* pada aspek spasial dan aspek temporal menggunakan fungsi *dist*. Fungsi *cbind* digunakan untuk menggabungkan kolom *longitude* dan kolom *latitude* menjadi sebuah matriks. Setelah itu mengubah tipe data “data_spasial dan data_temporal” menjadi matriks karena hasil *dist* berupa matriks segitiga atas sehingga perlu di normalisasi. Selanjutnya $n(row)$ digunakan untuk menghitung banyaknya baris yang terdapat pada data. *Clustering* dimulai dari objek pertama (o_1) di vektor *cv*. Objek o_2 di *D* dipilih setelah objek o_1 diproses. Jika objek o_1 belum termasuk ke dalam *cluster*, maka dicari irisan dari tetangga (o_i, Eps_1) dan tetangga (o_i, Eps_2) menggunakan fungsi *intersect* yang kemudian hasil irisannya dimasukkan ke dalam variabel *reachables*. Suatu objek dikatakan *noise* jika jumlah objek yang dikembalikan dalam *Eps-Neighborhood* kurang dari $MinPts$, dalam hal ini berarti o_i tidak memiliki cukup tetangga untuk membentuk sebuah *cluster*. *Eps-Neighborhood* adalah objek yang terjangkau secara spasial dan temporal oleh objek ke- i . Objek dengan label *noise* dapat berubah pada proses selanjutnya jika objek tidak *directly density-reachable* namun *density-reachable* dari objek-objek lain di *D*. *Directly density-reachable* merupakan objek yang terjangkau langsung oleh objek ke- i sedangkan *density-reachable* adalah objek yang terjangkau oleh tetangga dari objek ke- i (Gambar 7).



Gambar 6 *Directly density-reachable* dan *Density-reachable*

Implementasi algoritme dilakukan dengan memodifikasi fungsi dbscan yang terdapat pada *package* fpc. Potongan program modifikasi dbscan dapat pada Gambar 8.

```

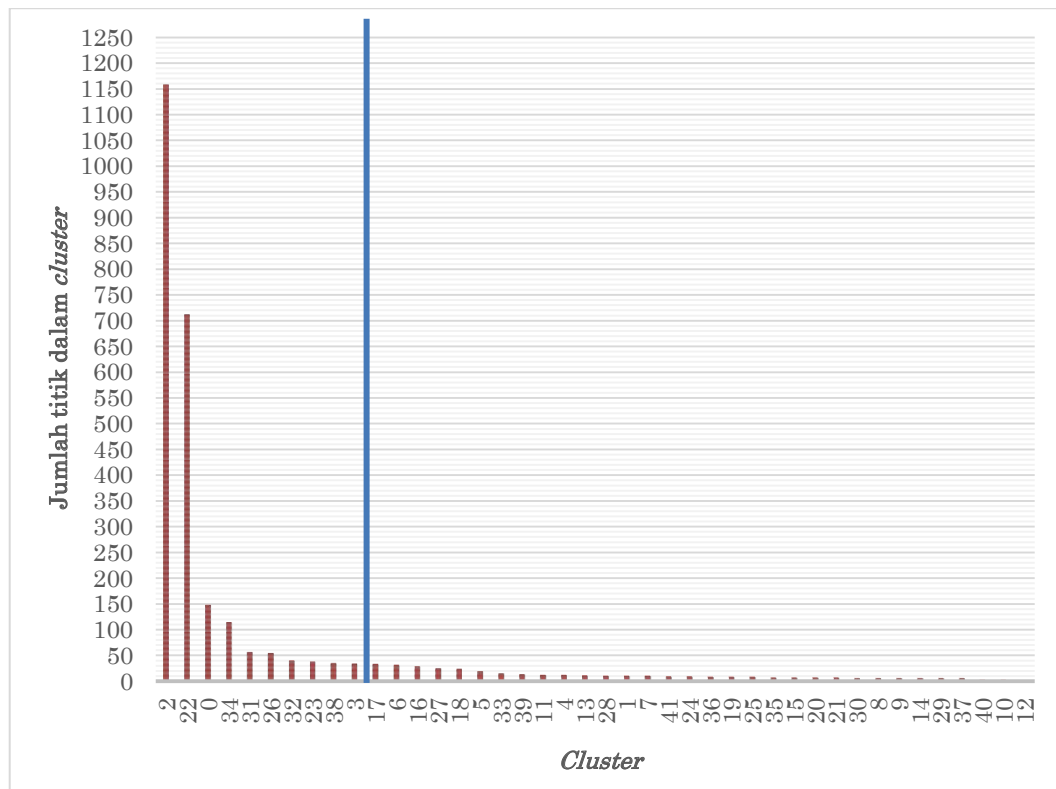
1 stdbscan2 = function (data, eps, eps2, MinPts = 7, seeds = TRUE,
2 countmode = NULL)
3 {
4 data_spatial<- dist(cbind(data$latitude,data$longitude))
5 data_temporal<- dist(cbind(data$date))
6
7 data_spatial <- as.matrix(data_spatial)
8 data_temporal <- as.matrix(data_temporal)
9 n <- nrow(data_spatial)
10
11 classn <- cv <- integer(n)
12 isseed <- logical(n)
13 cn <- integer(1)
14
15 for (i in 1:n) {
16   if (i %in% countmode)
17     cat("Processing point ", i, " of ", n, ".\n")
18   unclass <- (1:n)[cv < 1]
19     if (cv[i] == 0) {
20       reachables <- intersect(unclass[data_spatial[i, unclass]
21 <= eps], unclass[data_temporal[i, unclass] <= eps2])
22       if (length(reachables) + classn[i] < MinPts)
23         cv[i] <- (-1)
24       else {
25         cn <- cn + 1
26         cv[i] <- cn
27         isseed[i] <- TRUE
28         reachables <- setdiff(reachables, i)
29         unclass <- setdiff(unclass, i)
30         classn[reachables] <- classn[reachables] + 1
31         while (length(reachables)) {
32           cv[reachables] <- cn
33           ap <- reachables
34           reachables <- integer()
35           for (i2 in seq(along = ap)) {

```

Gambar 7 Kode Fragment dbscan

Clustering Menggunakan ST-DBSCAN

Proses *clustering* berhasil dijalankan dengan nilai tiap parameter yaitu, $Eps_1=0.2$, $Eps_2=7$ dan $MinPts=7$ serta menghasilkan 41 *cluster* dengan 712 *noise*. *Cluster* ini kemudian dibagi menjadi *cluster* yang bisa dianalisis dengan *cluster* yang tidak bisa dianalisis. Gambar 8 menunjukkan banyaknya titik per *cluster* dimana garis *abline* vertikal merupakan garis perpotongan antar *cluster* yang bisa dianalisis dengan yang tidak. *Cluster* yang dapat dianalisis merupakan *cluster* yang setidaknya memiliki minimum 30 *hotspot*, hal ini dikarenakan pola *cluster* sudah dapat dikategorikan tipe polanya menurut Pöelitz dan Andrienko (2010) sedangkan *cluster* yang memiliki jumlah *hotspot* kurang dari 30, pola *cluster* dari *cluster* tersebut tidak dapat terdeteksi. *Cluster* yang memiliki jumlah *hotspot* minimum 30 *hotspot* dapat dikatakan sebagai *cluster* besar.



Gambar 8 Jumlah titik per cluster

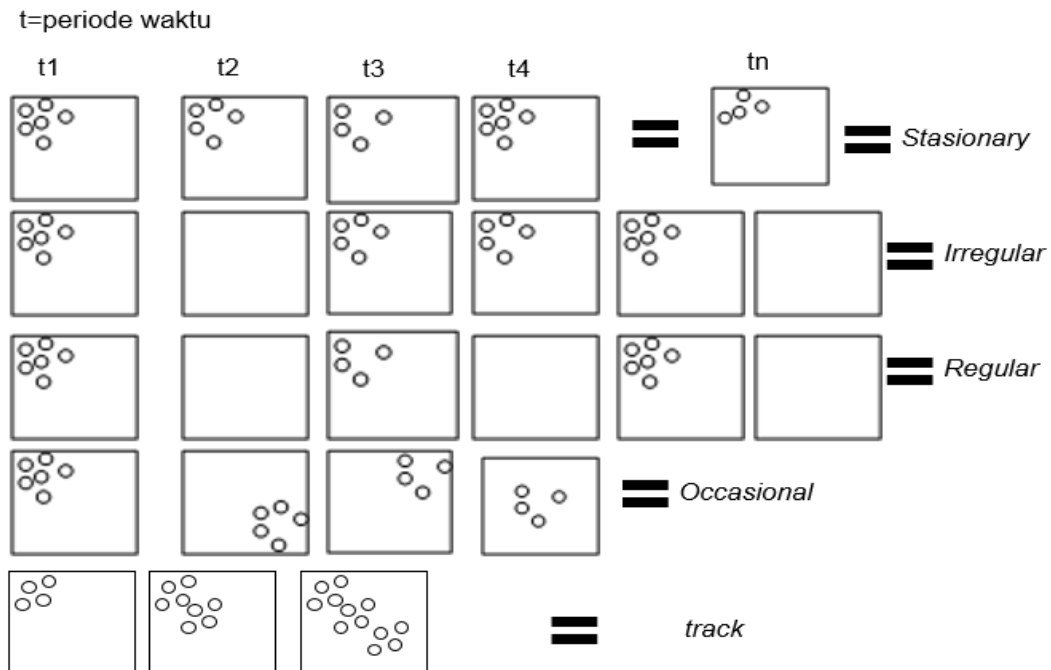
Analisis Cluster

Implementasi algoritme ST-DBSCAN dengan menggunakan bahasa pemrograman R yang diaplikasikan pada *dataset hotspot* di Sumatera Selatan dilakukan untuk menemukan pola kebakaran hutan. Pada hasil *cluster* didapatkan 13 *cluster* yang termasuk ke dalam *cluster* besar yang dapat dianalisis polanya. Tabel 2 menunjukkan *cluster-cluster* yang masuk ke dalam *cluster* besar.

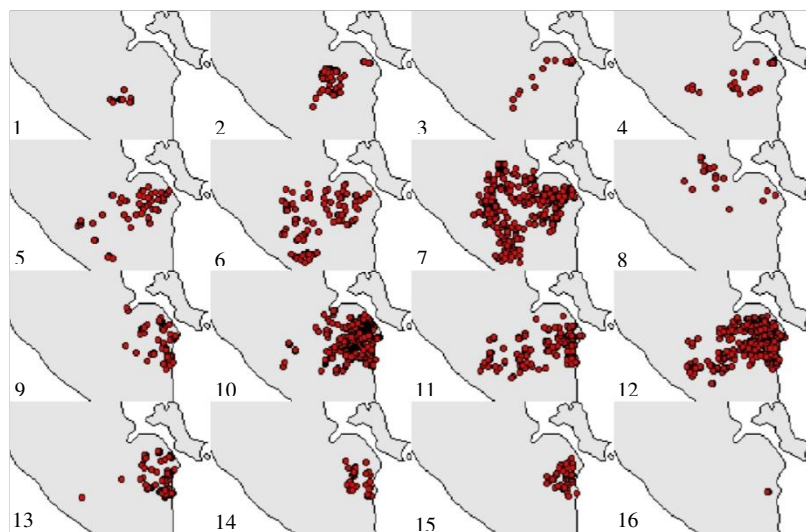
Tabel 4 Jumlah hotspot dalam cluster besar

Cluster ke-	Jumlah Hotspot
2	2029
22	1157
34	150
31	115
26	58
32	55
23	41
38	39
3	36
17	35
6	34
16	30

Cluster besar kemudian dianalisis polanya berdasarkan tipe pola data *spatio-temporal* menurut Pöelitz dan Andrienko (2010). Ilustrasi pola dapat dilihat pada Gambar 9.

Gambar 9 Ilustrasi pola *spatio-temporal* (Purwanto 2012)

Tiap *cluster* besar akan dibagi menjadi per periode, yaitu tujuh hari untuk perbandingan lokasi persebaran *hotspot* pada suatu periode dengan periode selanjutnya. Waktu 7 hari dipilih karena kondisi cuaca yang hampir mirip dan perubahan cuaca yang tidak signifikan. Sebagai contoh pada *cluster* 2 didapatkan 16 periode (Gambar 10).

Gambar 10 *Cluster* 2 pada periode 1 (kiri-atas) dan periode 16 (kanan-bawah)

Pada periode 7 *hotspot* muncul pada seluruh kabupaten kecuali kota. Selanjutnya pada periode 10 dan 12 *hotspot* terpusat pada Kabupaten Ogan Komering Ilir maka dapat dikatakan *cluster* 2 mempunyai pola *occasional*. Analisis 13 *cluster* besar pada penelitian ini menunjukkan bahwa tipe pola *spatio-temporal* yang paling banyak muncul adalah pola *stationary*. Terdapat 7 pola *stationary* yang muncul dan 4 diantaranya terdapat di kabupaten Muara Banyu Asin. Dengan adanya hasil analisis berupa pola kebakaran hutan diharapkan pemerintah dapat mengambil tindakan yang tepat untuk menangani kejadian kebakaran hutan. Sebagai contoh, untuk pola *stationary* maka dapat dilakukan pemberitahuan terhadap masyarakat sekitar dan pihak terkait untuk lebih waspada terhadap kemunculan kebakaran hutan agar dapat meminimalisir dampak yang terjadi.

SIMPULAN

Bahasa pemrograman R mampu mengimplementasikan algoritme ST-DBSCAN. Dengan mengimplementasikan algoritme ST-DBSCAN menggunakan R *runtime* yang didapatkan adalah 1.14 detik. Pada *dataset* Sumatera Selatan ditemukan 41 *cluster* dengan 13 *cluster* besar yang dapat dianalisis. Tipe pola *spatio-temporal* paling banyak ditemukan adalah tipe pola *stationary* yang banyak muncul di Kabupaten Musi Banyu Asin.

DAFTAR PUSTAKA

- Adinugroho WC, Suryadiputra INN, Saharjo BH, Siboro L. 2005. *Panduan Pengendalian Kebakaran Hutan dan Lahan Gambut*. Proyek Climate Change, Forests and Peatlands in Indonesia. Bogor: Wetlands International-Indonesia Programme dan Wildlife Habitat Canada.
- Birant D, Kut A .2007. *ST-DBSCAN: An Algorithm for Clustering Spatial-temporal data*. *Data and Knowledge Engineering*. 60:208-221.
- Gaonkar M N, Sawant K. 2013. *AutoEPsDBSCAN : DBSCAN with Eps Automatic for Large Datase*. 2:2319-2526.
- Pöelitz C dan Andrienko N. 2010. *Finding Arbitrary Shaped Cluster with Related Extents in Space and Time*. IEEE-VGTC Symposium on Visualization.
- Purwanto U Y, Barus B, dan Adrianto H A. 2012. *Spatial clustering of hotspot using DBSCAN and ST-DBSCAN*. Bogor (ID): Institut Pertanian Bogor.
- Rao K V, Govardhan A, dan Rao K V C. 2012. *Spatiotemporal Data Mining: Issues, Tasks and Applications*. Chennai (IN): Engg Journals Publications.
- Shekhar dan Chawla. 2003. *Spatial Databases A Tour*. New Jersey. Prentice Hall