



UNIVERSITAS INDONESIA

PENGEMBANGAN SIMULATOR LIMPASAN BERBASIS  
SISTEM INFORMASI GEOGRAFIS MENGGUNAKAN *HORTON*  
*INFILTRATION EXCESS MODEL*

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

FERI FRASTIANSYAH

0606072212

FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK SIPIL  
DEPOK  
JULI 2010



UNIVERSITY OF INDONESIA

DEVELOPMENT OF GIS BASED RUNOFF SIMULATOR  
USING HORTON INFILTRATION EXCESS MODEL

UNDERGRADUATE THESIS

Proposed as one of the requirement to obtain the Bachelor degree in Engineering

FERI FRASTIANSYAH

0606072212

FACULTY OF ENGINEERING  
CIVIL ENGINEERING STUDY PROGRAM

DEPOK

JULY 2010

## HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri,  
dan semua sumber baik yang dikutip maupun dirujuk  
telah saya nyatakan dengan benar.

Nama : Feri Frastiansyah

NPM : 0606072212

Tanda Tangan :

Tanggal : 6 Juli 2010



## STATEMENT OF ORIGINALTY

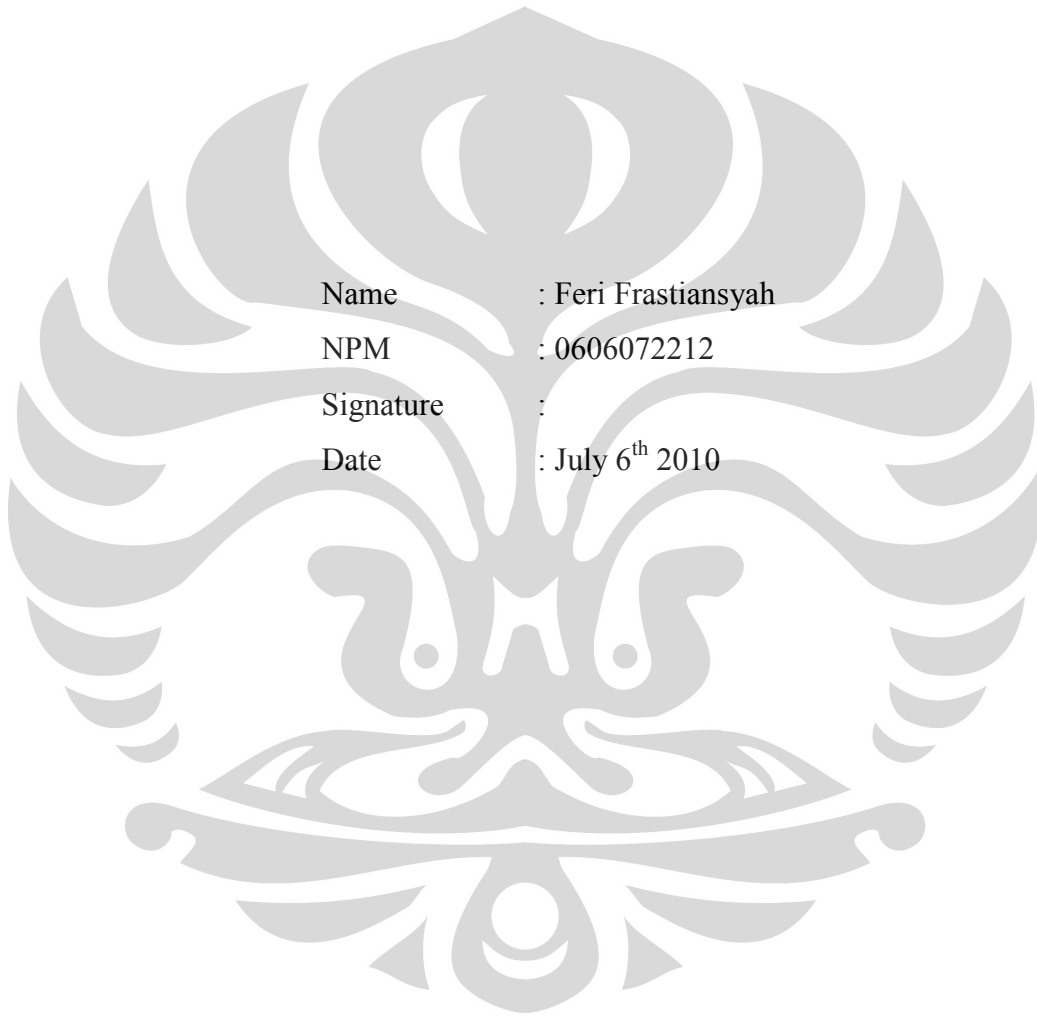
This undergraduate thesis authorized by my self,  
and all sources which referred or quoted  
had been write down correctly.

Name : Feri Frastiansyah

NPM : 0606072212

Signature :

Date : July 6<sup>th</sup> 2010



## HALAMAN PENGESAHAN

Skripsi ini diajukan oleh

Nama : Feri Frastiansyah  
NPM : 0606072212  
Program Studi : Teknik Sipil  
Judul Skripsi : Pengembangan Simulator Limpasan Berbasis  
Sistem Informasi Geografis Menggunakan *Horton  
Infiltration Excess Model*

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Sipil, Fakultas Teknik, Universitas Indonesia

### DEWAN PENGUJI

Pembimbing I : Ir. Herr Soeryantono, M.Sc, Ph.D (.....)  
Pembimbing II : Toha Saleh, M.Sc (.....)  
Penguji : Dr. –Ing. Ir. Dwita Sutjningsih (.....)  
Penguji : Dwinanti Rika M. ST. MT. (.....)

Ditetapkan di : Depok

Tanggal : 6 Juli 2010

## PAGE OF ENDORSEMENT

This final assignment is submitted by

Name : Feri Frastiansyah  
NPM : 0606072212  
Study Program : Civil Engineering  
Title : Development of GIS Based Runoff Simulator Using  
Horton Infiltration Excess Model

Has been successfully maintained in the face of examiners and accepted as part of the requirement necessary to obtain the Bachelor degree in Engineering, Civil Engineering Program, Faculty of Engineering, University of Indonesia.

### BOARD OF EXAMINER

Counselor I : Ir. Herr Soeryantono, M.Sc, Ph.D (.....)  
Counselor II : Toha Saleh, M.Sc (.....)  
Examiner : Dr. -Ing. Ir. Dwita Sutjningsih (.....)  
Examiner : Dwinanti Rika M. ST. MT. (.....)

Defined in : Depok

Date : July 6<sup>th</sup> 2010

## KATA PENGANTAR/UCAPAN TERIMA KASIH

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, saya dapat menyelesaikan skripsi ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Jurusan Teknik Sipil pada Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi saya untuk menyelesaikan skripsi ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

1. Bapak Ir. Herr Soeryantono, M.Sc, Ph.D selaku dosen pembimbing yang telah memberikan informasi dan arahan sejak awal pengerjaan skripsi ini dengan konsep-konsep dan pemahaman yang sangat mendalam;
2. Bapak Toha Saleh, MSc. selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan skripsi ini;
3. orang tua dan keluarga saya yang telah memberikan bantuan dukungan material dan moral,
4. sahabat yang telah banyak membantu saya dalam menyelesaikan skripsi ini.

Akhir kata, saya berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 6 Juli 2010

Feri Frastiansyah

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI  
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

---

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Feri Frastiansyah  
NPM : 0606072212  
Program Studi : Teknik Sipil  
Departemen : Teknik Sipil  
Fakultas : Teknik  
Jenis karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty-Free Right) atas karya ilmiah saya yang berjudul :

**Pengembangan Simulator Limpasan Berbasis Sistem Informasi Geografis Menggunakan *Horton Infiltration Excess Model*.**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok  
Pada tanggal : 6 Juli 2010

Yang menyatakan

(Feri Frastiansyah)



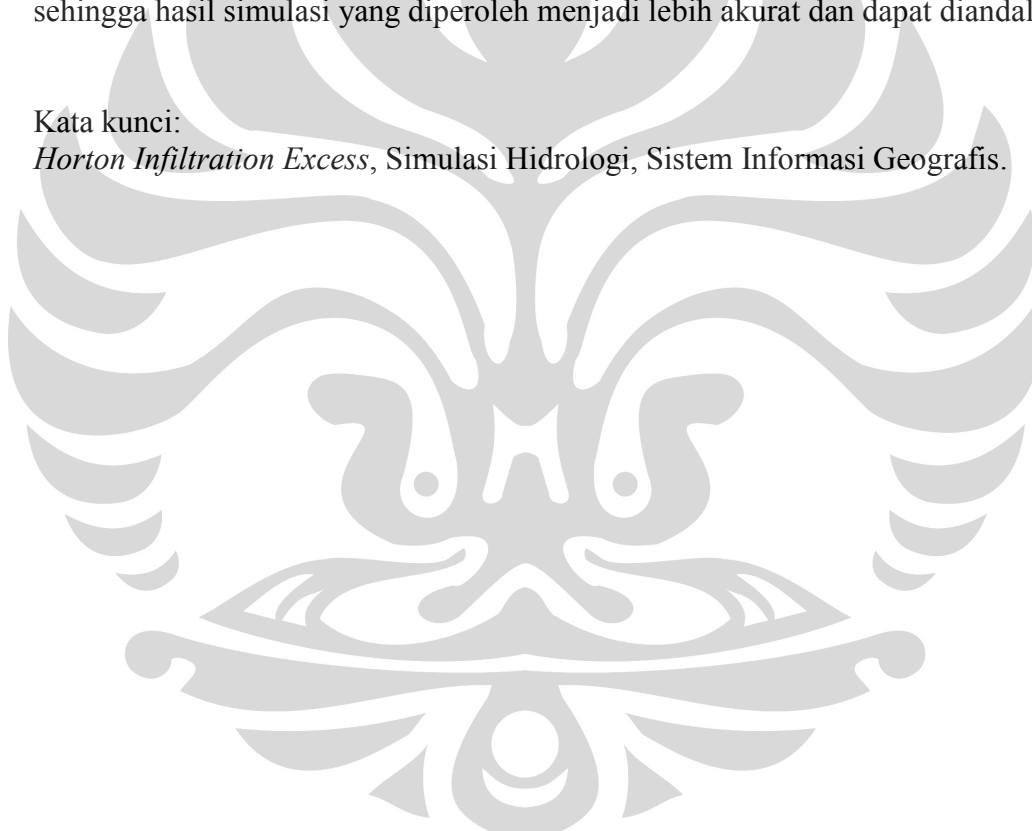
## ABSTRAK

Nama : Feri Frastiansyah  
Program Studi : Teknik Sipil  
Judul : Pengembangan Simulator Limpasan Berbasis Sistem Informasi Geografis Menggunakan *Horton Infiltration Excess Model*

*Horton Infiltration Excess* sebagai salah satu model simulasi hidrologi dalam menghitung besar limpasan permukaan, disadari masih bersifat klasik dan linear karena belum memperhitungkan fenomena fisik selain infiltrasi serta mengabaikan detail arah alirannya. Dengan menggunakan Sistem Informasi Geografis sebagai pengolah data spasial, model ini dikembangkan lebih lanjut sehingga hasil simulasi yang diperoleh menjadi lebih akurat dan dapat diandalkan.

Kata kunci:

*Horton Infiltration Excess*, Simulasi Hidrologi, Sistem Informasi Geografis.

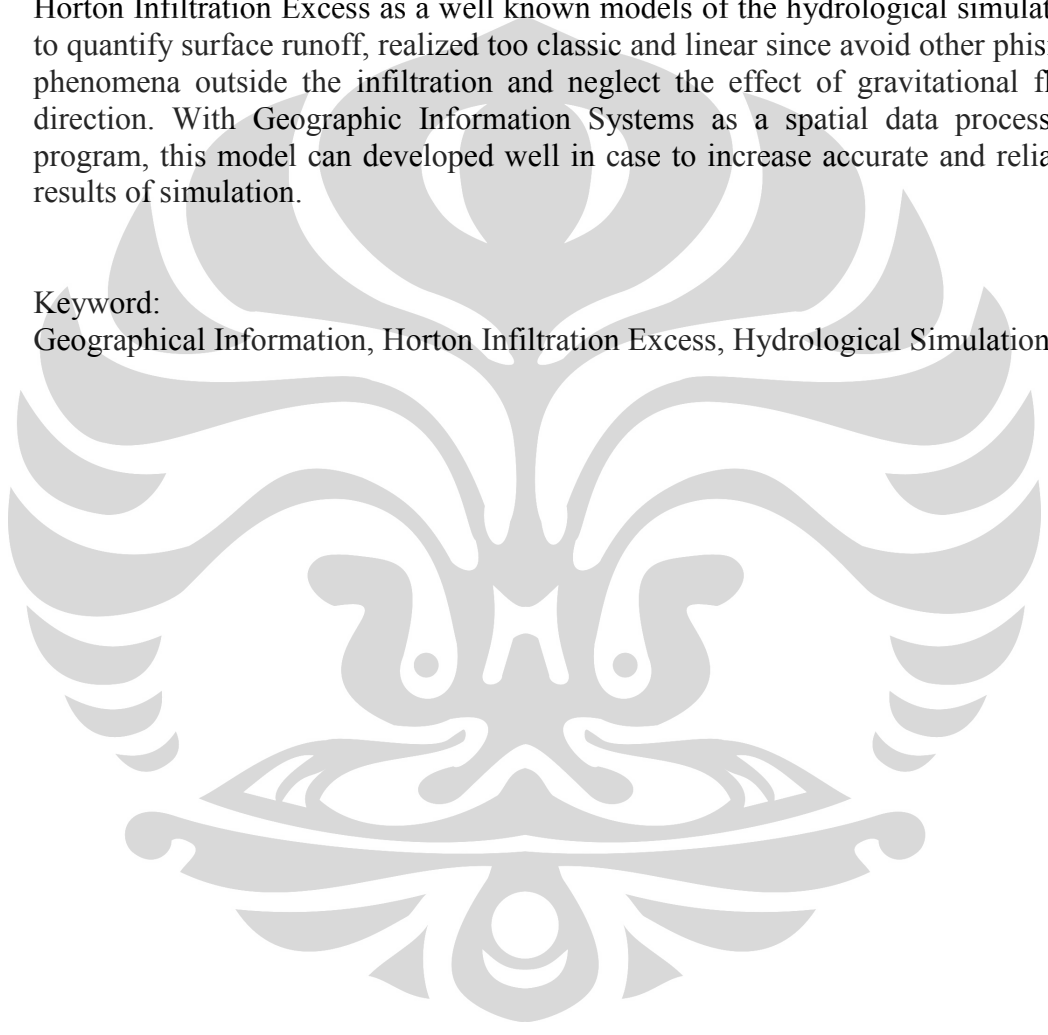


## ABSTRACT

Name : Feri Frastiansyah  
Study Program : Civil Engineering  
Title : Development of GIS Based Runoff Simulator Using Horton Infiltration Excess Model.

Horton Infiltration Excess as a well known models of the hydrological simulation to quantify surface runoff, realized too classic and linear since avoid other phisical phenomena outside the infiltration and neglect the effect of gravitational flow direction. With Geographic Information Systems as a spatial data processing program, this model can developed well in case to increase accurate and reliable results of simulation.

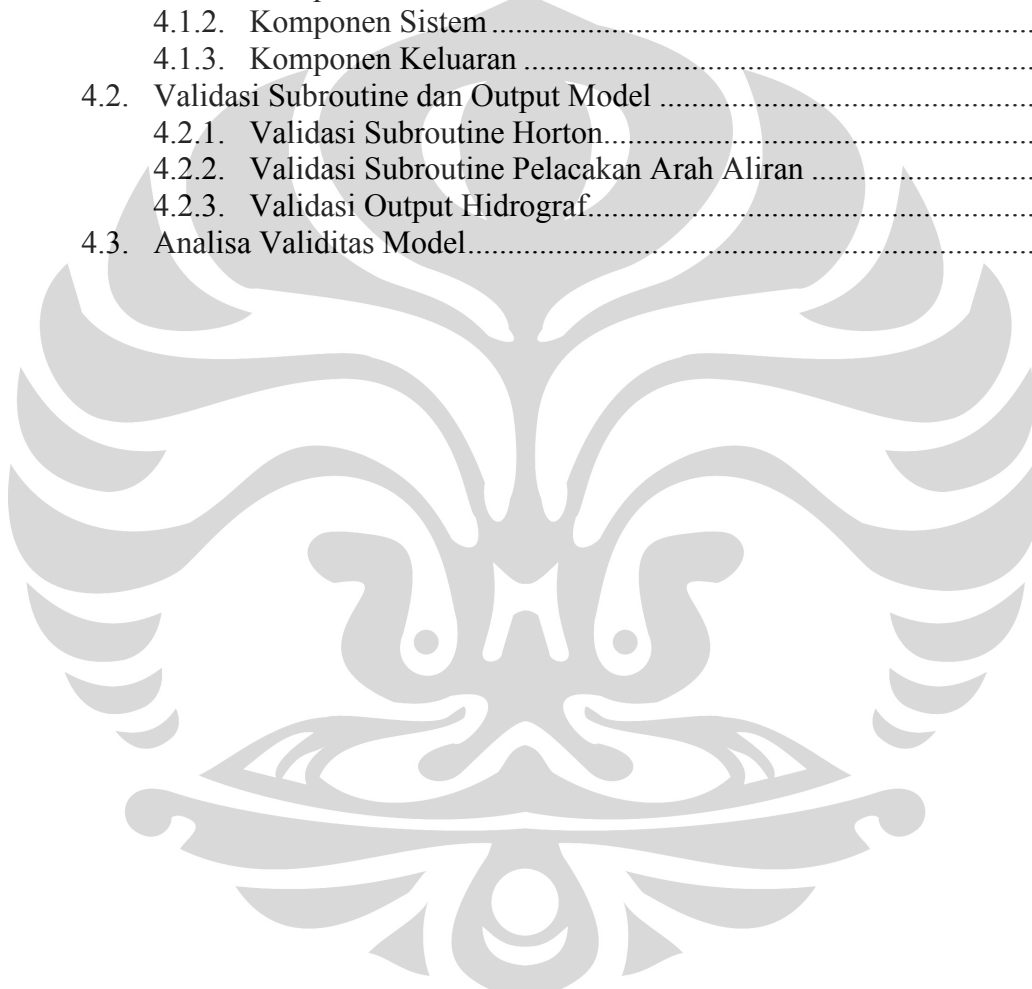
Keyword:  
Geographical Information, Horton Infiltration Excess, Hydrological Simulation.



## DAFTAR ISI

HALAMAN JUDUL.....	i
PERNYATAAN ORISINALITAS .....	iii
LEMBAR PENGESAHAN .....	v
KATA PENGANTAR .....	vii
LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH .....	ix
ABSTRAK .....	x
DAFTAR ISI.....	xi
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR .....	xiv
DAFTAR PUSTAKA .....	xx
<b>1. PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Tujuan Penulisan.....	3
1.3. Ruang Lingkup Pembahasan.....	3
1.4. Sistematika Penulisan .....	4
<b>2. METODE Kuantifikasi Aliran Limpasan dalam Siklus Hidrologi.....</b>	<b>6</b>
3.1. Siklus Hidrologi .....	6
3.2. Kuantifikasi Limpasan dengan Horton Infiltration Excess Model .....	8
2.2.1. Hujan Efektif.....	9
2.2.1.1. Analisis Frekuensi Hujan .....	10
2.2.1.2. Analisis Intensitas Hujan.....	12
2.2.1.3. Analisis Pola Sebaran Hujan.....	13
2.2.2. Infiltrasi .....	16
3.3. Kecepatan aliran limpasan permukaan.....	20
<b>3. APLIKASI SISTEM INFORMASI GEOGRAFIS DALAM Kuantifikasi Limpasan.....</b>	<b>22</b>
3.1. Sistem Informasi Geografis (SIG) .....	22
3.2. Komponen Utama SIG .....	23
3.3. Kemampuan Dasar SIG.....	25
3.3.1. Memetakan Letak.....	25
3.3.2. Memetakan Kuantitas.....	26
3.3.3. Memetakan Kerapatan .....	26
3.3.4. Memetakan Perubahan .....	26
3.4. Sumber Data dalam SIG.....	27
3.4.1. Peta Analog .....	27
3.4.2. Citra Satelit .....	27
3.4.3. Pengukuran Lapangan.....	28
3.4.4. Global Positioning Sistem (GPS).....	28
3.4.5. Foto Udara.....	28
3.5. Data Spasial dalam SIG .....	29
3.5.1. Model Data Raster.....	30

3.5.2.	Model Data Vektor.....	35
3.5.3.	Perbandingan Model Data Raster dan Vektor.....	37
3.6.	Kuantifikasi Limpasan Permukaan dengan menggunakan SIG.....	38
3.6.1.	<i>PreGIS Processing</i> .....	40
3.6.2.	<i>GIS Processing</i> .....	40
3.6.3.	<i>PostGIS Processing</i> .....	41
3.7.	Metodologi Pelaksanaan Kajian.....	43
<b>4.</b>	<b>PENGEMBANGAN DAN VALIDASI MODEL .....</b>	<b>44</b>
4.1.	Struktur Program.....	44
4.1.1.	Komponen Masukkan .....	46
4.1.2.	Komponen Sistem .....	50
4.1.3.	Komponen Keluaran .....	54
4.2.	Validasi Subroutine dan Output Model .....	54
4.2.1.	Validasi Subroutine Horton.....	55
4.2.2.	Validasi Subroutine Pelacakan Arah Aliran .....	57
4.2.3.	Validasi Output Hidrograf.....	58
4.3.	Analisa Validitas Model.....	60



## DAFTAR TABEL

Tabel 2.1. Harga Koefisien Intersepsi ( $C_{ij}$ ).....	10
Tabel 2.2. Harga Reduced Standar Deviation ( $\sigma_N$ ).....	11
Tabel 2.3. Harga reduced mean ( $Y_N$ ).....	12
Tabel 2.4. Harga reduced Variated ( $Y_T$ ).....	12
Tabel 2.5. Harga koefisien $f_0$ , $f_c$ , dan $k$ .....	18
Tabel 3.1. Perbandingan Struktur Data Vektor dan Raster.....	38
Tabel 4.1. Klasifikasi Kode Jenis Tanah.....	49
Tabel 4.2. Harga koefisien $f_0$ , $f_c$ , dan $k$ .....	50



## DAFTAR GAMBAR

Gambar 2.1. Siklus Hidrologi .....	6
Gambar 2.2. Proses Intersepsi .....	9
Gambar 2.3. Contoh Garis-garis Isohyet.....	15
Gambar 2.4. Perubahan Lengkung Laju Infiltrasi.....	19
Gambar 3.1. Klasifikasi Model Data Spasial .....	30
Gambar 3.2. Struktur Model Data Raster.....	31
Gambar 3.3. Struktur Penyimpanan Model Data Raster.....	31
Gambar 3.4. Poligon dalam Berbagai Macam Ukuran Sel .....	32
Gambar 3.5. Foto Udara (Raster) Sebagai Latar dari Layer Jalan (Vektor) .....	33
Gambar 3.6. Data Raster dalam Memodelkan Permukaan Bumi .....	34
Gambar 3.7. Data Raster dalam Mengklasifikasi Data Tutupan Lahan.....	35
Gambar 3.8. Representasi Data Titik .....	35
Gambar 3.9. Representasi Data Garis .....	36
Gambar 3.10. Representasi Data Area .....	36
Gambar 3.11. Proses Kuantifikasi Limpasan Permukaan dengan SIG.....	39
Gambar 3.12. Proses Utama Dalam Kajian .....	42
Gambar 4.1 Tampilan Utama .....	44
Gambar 4.2 Melakukan Import Raster Map .....	45
Gambar 4.3 Pengklasifikasian Data Tanah .....	45
Gambar 4.4 Struktur Penyimpanan Data Elevasi.....	46
Gambar 4.5 Rangkaian Sistem Simulasi.....	51
Gambar 4.6 Sistem Simulasi .....	51
Gambar 4.7 Sistem Simulasi Hujan-Alir .....	52
Gambar 4.8 Pelacakan Aliran Limpasan.....	53
Gambar 4.9 Kondisi Pelacakan yang Bermasalah .....	53
Gambar 4.10 Kondisi Pelacakan yang Bermasalah .....	53
Gambar 4.11 Hasil Pengujian Skenario 1 .....	55
Gambar 4.12 Hasil Pengujian Skenario 2 .....	55
Gambar 4.13 Hasil Pengujian Skenario 3 .....	56
Gambar 4.14 Kontur Pengujian Penacakan Aliran .....	57
Gambar 4.15 Hasil Pelacakan Arah Aliran .....	57
Gambar 4.16 Hasil Akhir Hidrograf .....	58
Gambar 4.17 Perubahan Parameter Bentuk DAS .....	59
Gambar 4.18 Perbandingan Hidrograf Akibat Bentuk DAS.....	59
Gambar 4.19 Perubahan Parameter Penutup Lahan dan Tata Guna Lahan .....	59
Gambar 4.20 Perbandingan Hidrograf Akibat Bentuk DAS.....	60

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Laju pembangunan yang tidak terkontrol dalam suatu kawasan tentunya akan membawa efek negatif pada proses hidrologi dalam kawasan tersebut. Perubahan semacam inilah yang nantinya akan mengakibatkan peningkatan volume limpasan permukaan yang dapat menyebabkan banjir. Perubahan tersebut dapat pula mengganggu kestabilan kuantitas air tanah yang dapat mengakibatkan dampak lain seperti penurunan permukaan tanah, intrusi air laut, dan lainnya.

Oleh karena itu, komponen hidrologi dalam suatu kawasan perlu dikelola secara bijak dengan pendekatan terpadu dan menyeluruh. Terpadu mencerminkan keterkaitan dengan berbagai aspek, berbagai pihak dan berbagai disiplin ilmu. Menyeluruh mencerminkan cakupan yang sangat luas, melintasi batas antar sumber daya, antar lokasi, hulu dan hilir, antar kondisi, jenis tata guna lahan, antar berbagai aspek dan antar multi disiplin.

Untuk melakukan pengelolaan komponen hidrologi, tentunya diperlukan sebuah penyelidikan awal berupa pemetaan dari berbagai data yang terkait. Penggunaan peta tematik yang terpisah-pisah menjadikan proses ini kurang efektif, sehingga dirasakan sulit untuk menggabungkan beberapa informasi yang berbeda dari masing-masing peta tersebut. Seiring dengan perkembangan komputer yang begitu pesat, kini telah tersedia sebuah alat bantu (*tools*) untuk menggabungkan berbagai informasi yang berbeda dalam peta analog dengan menggunakan Sistem Informasi Geografis (SIG) yang disebut peta digital.

Dengan menggunakan SIG, data yang disimpan akan menjadi lebih padat karena tersedia dalam bentuk digital. Menejemen penyimpanan data yang baik akan memudahkan penggabungan informasi geospasial yang telah ada, sehingga data dapat langsung digabungkan dalam waktu yang relatif singkat dan hasil dari penggabungan tersebut dapat langsung dilakukan

analisa spasial secara cepat, akurat, dan model analisa yang dapat dikembangkan sesuai dengan spesifikasi yang dibutuhkan.

Dalam kajian sebelumnya, telah dilakukan digitasi dari berbagai peta tematik yang terkait dengan pengelolaan komponen hidrologi dalam wilayah studi (DAS Sugutamu), yaitu peta topografi, peta geologi, peta sebaran hujan, peta *soil group*, dan peta tata guna lahan. (Rizaldy, 2009) Namun kajian ini belum mencakup kuantifikasi proses-proses hidrologi dalam kawasan tersebut, sehingga perlu dilakukan pengembangan lebih lanjut mengenai kuantifikasi proses-proses hidrologi dalam DAS Sugutamu dengan menggunakan SIG.

Untuk melakukan pemodelan dan simulasi proses hidrologi, bukanlah merupakan suatu proses yang mudah. Berbagai penelitian mengenai hal tersebut telah banyak dilakukan, baik secara *single-process* seperti *infiltration excess model* (Horton, 1939), *topographically included saturation excess model* (Beven and Kirkby, 1979), *saturation excess caused by impaired drainage or rapid interflow Model* (Hewlett dan Nutter, 1970); Maupun secara *multiple-process* seperti *soil moisture-based runoff model* (James A. Zollweg, 1995).

Hasil dari penelitian-penelitian tersebut kini telah banyak dipakai dan dikembangkan menjadi berbagai perangkat lunak (*software*) yang dapat mensimulasikan proses hidrologi dalam suatu kawasan, baik perangkat lunak yang berdiri sendiri (*Stand-Alone*), maupun yang dapat diintegrasikan kedalam Sistem Informasi Geografis. Sebuah perangkat SIG yang baik, biasanya sangat terbuka dalam menyediakan *Application Programming Interface*, sehingga para penggunanya dapat membuat dan menambahkan sendiri berbagai perangkat tambahan (*tools extension*) yang sesuai dengan kriteria dan kebutuhan analisa yang mereka harapkan.

Untuk mensimulasikan proses hidrologi kawasan menggunakan SIG, tentunya memerlukan beberapa perangkat tambahan khusus yang mampu melakukan simulasi dan kuantifikasi berdasarkan data yang dimasukkan kedalam SIG. Berbagai perangkat tambahan yang memiliki kemampuan tersebut tentunya telah banyak dikembangkan sebelumnya. Namun berbagai



perangkat tambahan tersebut umumnya memiliki harga yang relatif mahal, sehingga sering kali menyulitkan pengguna dari kalangan akademisi dan organisasi dalam mencari sebuah perangkat tambahan yang gratis dengan hasil yang dapat diandalkan. Oleh karena itu dibutuhkan suatu perangkat tambahan non-komersial yang bersifat *public-domain*, sehingga nantinya perangkat ini dapat digunakan secara luas oleh berbagai kalangan dan organisasi.

## 1.2. Tujuan Penulisan

Tujuan penulisan ini adalah membuat suatu perangkat tambahan (*tools extension*) berbasis Sistem Informasi Geografis yang mampu mensimulasi dan mengkuantifikasi proses hidrologi suatu kawasan dengan menggunakan resolusi tinggi dalam jumlah yang masif meliputi debit resapan (infiltrasi), debit aliran permukaan (*surface runoff*), dan debit aliran sungai dalam suatu DAS yang dikembangkan berdasarkan *infiltration excess model* (Horton, 1939).

## 1.3. Ruang Lingkup Pembahasan

Untuk menghindari pelebaran topik yang tidak sesuai dengan tujuan utama yang ingin dicapai, maka perlu dilakukan pembatasan ruang lingkup sesuai dengan tujuan utama penulisan ini, yaitu:

- Peristiwa fisik siklus hidrologi yang akan dimodelkan mencakup: hujan, infiltrasi, dan *surface-runoff*. Peristiwa fisik lain diluar peristiwa fisik tersebut tidak diperhitungkan;
- Aliran *surface-runoff* dimodelkan mengalir secara gravitasi berdasarkan keadaan topografi, bentuk pengaliran lain berupa saluran drainase buatan tidak diperhitungkan;
- Tingkat saturasi tanah permukaan dihitung secara *single-process*, fenomena pengurangan atau penambahan tingkat saturasi akibat efek

aliran air di bawah permukaan tanah, perkolasi dan kapilaritas tidak diperhitungkan;

- Badan air seperti danau, situ, empang, waduk, dan badan air lainnya di luar sungai tidak diperhitungkan, sehingga efek *depression storage* diasumsikan tidak terjadi;
- Pemetaan tata guna lahan dibuat secara hipotetikal berdasarkan hasil pengindraan jarak jauh (*remote sensing*) berupa citra satelit yang telah disediakan; dan
- Pemetaan jenis tanah dan nilai konduktifitas kejenuhan (*saturated conductivity*) dibuat secara hipotetikal, tanpa melalui hasil pengujian lapangan dan laboratorium.
- Metode validasi akan dilakukan dengan cara membandingkan hasil dari model komputer yang akan dibuat dengan perangkat lunak winTR-20.

#### 1.4. Sistematika Penulisan

Secara garis besar, sistematika penyusunan yang digunakan dalam tulisan ini dibagi menjadi enam bab, yaitu: Pendahuluan, Aliran Limpasan dalam Perspektif Sistem Informasi Geografis, Pengembangan dan Validasi Model Komputer, Penerapan Pada Wilayah Studi, dan Kesimpulan.

Pada bab I, Pendahuluan, akan dijelaskan mengenai latar belakang, tujuan penulisan, ruang lingkup pembahasan, metode penulisan, dan sistematika penulisan yang akan dilakukan.

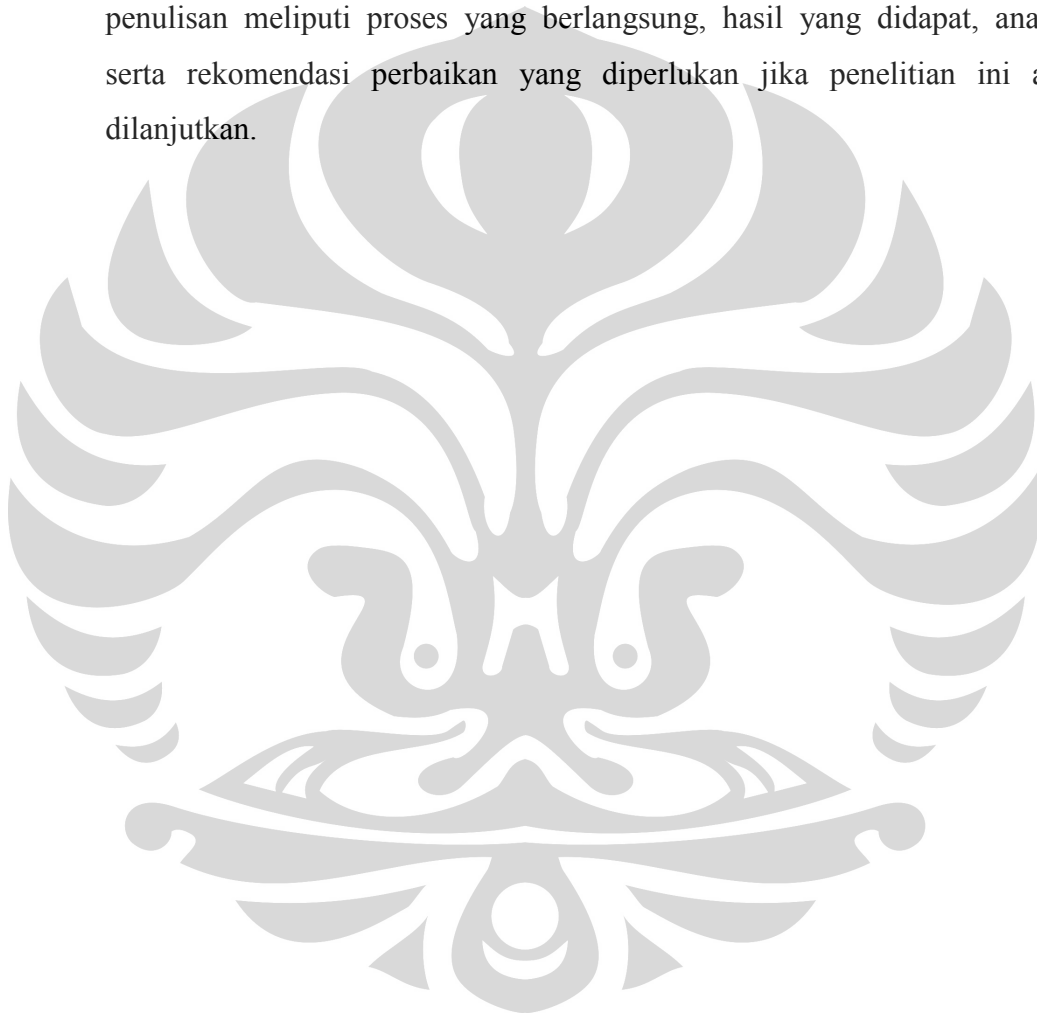
Pada bab II, Metode Kuantifikasi Aliran Limpasan dalam Siklus Hidrologi, akan dijelaskan mengenai hasil sintesa dari berbagai studi literatur yang berkaitan dengan peristiwa-peristiwa hidrologi yang terjadi dalam suatu DAS, yang kemudian diformulasikan menjadi suatu model matematik yang dapat diterapkan dalam perhitungan komputer serta proses-proses yang akan dilakukan pada saat penulisan.

Pada bab III, Aplikasi Sistem Informasi Geografis Dalam Kuantifikasi Limpasan, akan dijelaskan mengenai perangkat Sistem

Informasi Geografis serta pengembangan model komputer berdasarkan model matematik yang telah dikembangkan sebelumnya pada bab II.

Pada bab IV, Pengembangan dan Validasi Model, akan dijelaskan mengenai struktur program yang akan dibuat, cara kerja program, parameter masukan dan keluaran program, serta proses validasi berbagai algoritma sampai model siap digunakan.

Pada bab V, Kesimpulan, berisi kesimpulan yang didapat dari hasil penulisan meliputi proses yang berlangsung, hasil yang didapat, analisa, serta rekomendasi perbaikan yang diperlukan jika penelitian ini akan dilanjutkan.

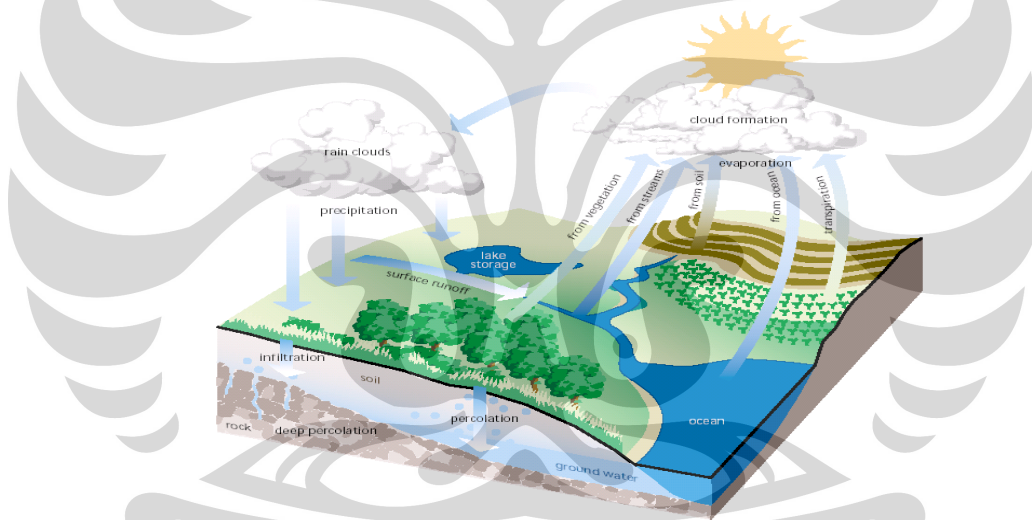


## BAB II

### METODE KUANTIFIKASI ALIRAN LIMPASAN DALAM SIKLUS HIDROLOGI

#### 2.1. Siklus Hidrologi

Daur atau siklus hidrologi secara sederhana dapat didefinisikan sebagai pergerakan air dari permukaan laut, kemudian jatuh ke permukaan tanah dan akhirnya mengalir kembali ke laut. Dalam proses daur tersebut terdapat berbagai kejadian dan faktor kondisi alam yang saling mempengaruhi satu sama lainnya, sehingga menyebabkan siklus tersebut menjadi suatu fenomena alam yang kompleks.



**Gambar 2.1.** Siklus Hidrologi (sumber: *Stream Corridor Restoration*)

Secara lebih detail siklus hidrologi dapat dijelaskan sebagai berikut: akibat panas yang bersumber dari matahari, terjadi proses penguapan atau evaporasi, baik dari permukaan air laut, air sungai, penguapan dari permukaan tanaman atau transpirasi, serta dari permukaan tanah. Pada ketinggian tertentu, uap air ini akan diubah menjadi awan. Apabila kondisi alam memungkinkan, akan terjadi presipitasi yang dapat berupa hujan, salju, es dan sebagainya. Sebagian kecil dari air ini akan diuapkan kembali sebelum sampai ke tanah, dan selebihnya merupakan hujan. Air yang jatuh ke permukaan tanah, sebagian akan mengalir ke laut atau sungai, dan

sebagian lainnya mengalir sebagai limpasan bawah permukaan serta meresap ke dalam tanah atau infiltrasi.

Dalam siklus hidrologi dikenal terdapat dua macam siklus, yaitu siklus panjang dan siklus pendek. Siklus pendek merupakan perjalanan air yang berlangsung sangat singkat dimana air hujan yang jatuh langsung kembali ke laut. Sedangkan siklus panjang dapat terjadi dalam dua kondisi, yaitu kondisi diatas permukaan dan kondisi dibawah permukaan. Saat air yang jatuh langsung mengalir kembali laut dalam bentuk aliran limpasan permukaan (*surface runoff*) melalui sungai, terjadilah siklus panjang pada kondisi diatas permukaan, sedangkan kondisi dibawah permukaan terjadi saat air yang jatuh tidak langsung mengalir ke laut melainkan terlebih dahulu meresap ke dalam tanah (infiltrasi) dan terus turun kebawah (perkolasi) sampai ke lapisan jenuh air (*saturated zone*) dan bergerak perlahan-lahan menuju ke sungai atau sampai langsung ke laut dalam bentuk aliran bawah permukaan (*subsurface runoff*).

Dari uraian diatas dapat terlihat bahwa terdapat empat macam proses utama dalam siklus hidrologi yang memegang peranan yang sangat penting, yaitu presipitasi, evaporasi, infiltrasi, serta limpasan permukaan (*surface runoff*) dan limpasan bawah permukaan (*subsurface runoff*). Untuk dapat mengkuantifikasi limpasan permukaan, diperlukan pemahaman terhadap peristiwa-peristiwa fisik tersebut agar suatu rangkaian masukan (*input*), berupa tinggi hujan, dapat di proses menjadi suatu rangkaian keluaran (*output*), berupa tinggi aliran atau debit aliran limpasan, melalui suatu proses pengalihragaman (*transformation*) tertentu.

Berbagai model proses pengalihragaman hujan-limpasan ini telah banyak di teliti dan kembangkan sebelumnya. Salah satu model yang paling populer dan telah banyak dipergunakan adalah *Horton infiltration excess model* (Horton, 1939) yang selanjutnya akan digunakan sebagai model utama yang akan dikembangkan menjadi model komputer yang akan dibuat pada tulisan ini. Uraian lebih lanjut mengenai metode ini dan dibahas pada sub-bab berikutnya.

## 2.2. Kuantifikasi Limpas dengan *Horton Infiltration Excess Model*

Dalam siklus hidrologi, limpasan permukaan (*surface runoff*) merupakan salah satu komponen yang paling penting dan terkadang menimbulkan banyak masalah jika terjadi dalam jumlah yang cukup banyak. Oleh karena itu, perhitungan limpasan permukaan sangat diperlukan untuk menghasilkan suatu *engineering design* yang dapat mencegah timbulnya masalah-masalah terkait limpasan hujan.

Horton (1939, dalam Schwab, G.O, and R.K. Frevert, 1981) secara teoritis mendefinisikan besarnya limpasan permukaan dapat hitung dengan mengurangi besaran hujan dengan besaran infiltrasi sehingga didapatkan sisa (residual) dari infiltrasi yang kemudian mengalir secara gravitasi. Selain dipengaruhi oleh besaran infiltrasi, besaran limpasan permukaan juga dipengaruhi oleh kehilangan air hujan akibat pepohonan (*vegetal cover*), ditambah dengan *stemflow*, *throughfall*, dan faktor-faktor fisik lain yang mempengaruhinya. Sehingga secara matematis besarnya aliran limpasan dapat dihitung dengan menggunakan rumus:

$$D_r = D_p - D_{li} - D_i - D_s - D_e \quad (2.1)$$

dimana:  $D_r$  = limpasan permukaan (*surface runoff*) (mm/det)

$D_p$  = hujan (mm/det)

$D_{li}$  = kehilangan awal (*interception*) (mm/det)

$D_i$  = kehilangan akibat infiltrasi (mm/det)

$D_s$  = *depression storage* (mm/det)

$D_e$  = kehilangan akibat transpirasi dan evaporasi (mm/det).

Dalam kajian ini efek *depression storage* (seperti waduk, danau, dll) diasumsikan tidak terjadi sehingga  $D_s = 0$ . Saat hujan terjadi, matahari biasanya tertutup oleh awan-awan pekat yang menyebabkan suhu dan kelembapan udara menjadi cukup rendah, sehingga kehilangan akibat transpirasi dan evaporasi akan menjadi sangat kecil dan dapat diabaikan, sehingga  $D_e = 0$ . Sedangkan tinggi hujan dikurangi kehilangan awal akibat intersepsi ( $D_p - D_{li}$ ) merupakan tinggi hujan efektif ( $D_{pe}$ ) yang didefinisikan sebagai tinggi hujan sesungguhnya yang langsung melakukan kontak dengan

tanah. Sehingga rumusan awal dari horton tersebut dapat disederhanakan menjadi rumus berikut:

$$D_r = D_{pe} - D_i \quad (2.2)$$

dimana:  $D_r$  = limpasan permukaan (mm/det)

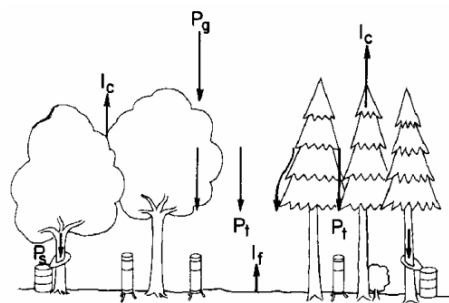
$D_{pe}$  = hujan efektif (mm/det)

$D_i$  = kehilangan akibat infiltrasi (mm/det)

Rumusan inilah yang nantinya akan digunakan sebagai rumus dasar yang digunakan dalam pengembangan model komputer untuk mengkuantifikasi aliran limpasan permukaan yang akan dibuat pada penulisan ini. Uraian lebih lanjut mengenai hujan efektif dan infiltrasi akan dibahas lebih mendalam pada sub-bab berikutnya.

### 2.2.1. Hujan Efektif

Seperti yang telah dijelaskan sebelumnya, hujan efektif ( $D_{pe}$ ) merupakan tinggi hujan sesungguhnya yang langsung melakukan kontak dengan tanah. Besarnya hujan efektif sangat bergantung kepada jenis dan kerapatan vegetasi yang ada pada suatu kawasan. Hal ini tentu saja sangat mempengaruhi besarnya aliran limpasan, karena sebelum hujan sampai ke permukaan tanah, terlebih dahulu hujan berinteraksi dengan vegetasi yang ada, sehingga akan mengurangi jumlah dan waktu sampai hujan tersebut ke permukaan tanah. Interaksi yang mungkin saja terjadi pada saat hujan jatuh adalah berintersepsi dengan kanopi vegetasi yang ada (pohon, semak, atau rumput), lalu beberapa pecah dan kembali ke udara, tertahan pada daun, jatuh langsung, atau menetes dan mengalir perlahan melalui batang menuju ke permukaan tanah.



**Gambar 2.2.** Proses Intersepsi (sumber: *After Hewlett 1982*)

Interaksi inilah yang membuat besaran hujan yang sesungguhnya sampai ke permukaan tanah tidak sepenuhnya sama dengan hujan yang jatuh dari awan, sehingga faktor-faktor seperti ini harus di perhitungkan dalam mengkuantifikasi besarnya aliran limpasan yang terjadi. Untuk menghitung besarnya hujan efektif, dapat dihitung dengan menggunakan rumus semi-empiris yang di gagas oleh Kenneth W. Tate, *Department of Agronomy & Range Science*, sebagai berikut:

$$D_{pe} = C_{li} I \quad (2.3)$$

dimana:  $D_{pe}$  = hujan efektif (mm/detik)

$C_{li}$  = koefisien intersepsi

$I$  = intensitas hujan (mm/detik)

Besarnya koefisien intersepsi ( $C_{li}$ ) ditentukan dengan menggunakan tabel empiris berikut:

**Tabel 2.1.** Harga Koefisien Intersepsi ( $C_{li}$ )

Vegetasi	Koefisien Intersepsi
Hutan Kerapatan Tinggi	0.07 – 0.13
Hutan Kerapatan Sedang	0.15 – 0.30
Hutan Kerapatan Rendah	0.35 – 0.60
Lahan Perkebunan	0.60 – 0.75
Lahan Persawahan	0.70 – 0.85
Lahan dengan rumput	0.80 – 0.90
Tanpa Vegetasi	0.90 – 1.00

*Sumber: Rangeland Watershed Program*

Sedangkan besarnya intensitas dan pola sebaran yang terjadi, dapat dihitung dengan menggunakan analisis frekuensi, analisis intensitas, dan analisis pola sebaran hujan yang akan dijelaskan pada sub-bab selanjutnya.

#### 2.2.1.1. Analisis Frekuensi Hujan

Sistem hidrologi terkadang dipengaruhi oleh peristiwa-peristiwa yang ekstrem, seperti hujan lebat, banjir, dan kekeringan. Tujuan dari analisis frekuensi data hidrologi yaitu berkaitan dengan besaran peristiwa-peristiwa ekstrem yang berkaitan dengan frekuensi kejadiannya melalui penerapan distribusi kemungkinan. Analisa frekuensi adalah analisa yang dilakukan untuk menentukan atau



memperkirakan kejadian curah hujan berdasarkan masa ulang peristiwa yang dapat diharapkan menyamai atau lebih besar dari pada rata-rata curah hujan. Analisa frekuensi yang digunakan berdasarkan metode Gumbel, dengan cara analitis.

- rata-rata curah hujan

$$\bar{X} = \frac{\sum X}{N} \quad (2.4)$$

- standar deviasi  $\sigma_x$

$$S_X = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{(n-1)}} \quad (2.5)$$

- rumus Gumble

$$X_{Tr24} = \bar{X} + \frac{\sigma_X}{\sigma_N} (Y_T - Y_N) \quad (2.6)$$

dimana:  $X_{Tr24}$  = nilai curah hujan pada periode ulang  $Tr$  tahun

$\bar{X}$  = nilai rata-rata curah hujan pada data tinjauan (mm)

$S_X$  = standar deviasi

$S_N$  = nilai reduksi standar deviasi pada  $N$  tahun seri data

$Y_T$  = nilai reduksi variasi berdasarkan lama periode ulang

$Y_N$  = nilai reduksi rata-rata

Nilai  $\sigma_N$  pada rumus 2.6 didapat dengan menggunakan tabel berikut sesuai dengan jumlah seri tahun data yang diperoleh:

**Tabel 2.2.** Harga Reduced Standar Deviation ( $\sigma_N$ )

M	0	1	2	3	4	5	6	7	8	9
10	0.949	0.967	0.983	0.997	1.009	1.020	1.031	1.041	1.049	1.056
20	1.062	1.069	1.075	1.081	1.086	1.091	1.096	1.100	1.104	1.108
30	1.112	1.115	1.119	1.122	1.125	1.128	1.131	1.133	1.136	1.138
40	1.141	1.143	1.145	1.148	1.149	1.151	1.153	1.155	1.157	1.159
50	1.160	1.162	1.163	1.165	1.166	1.168	1.169	1.170	1.172	1.173
60	1.174	1.175	1.177	1.178	1.179	1.180	1.181	1.182	1.183	1.184
70	1.185	1.186	1.187	1.188	1.189	1.189	1.190	1.191	1.192	1.193
80	1.193	1.194	1.195	1.195	1.196	1.197	1.198	1.198	1.199	1.200
90	1.200	1.201	1.202	1.202	1.203	1.203	1.204	1.204	1.205	1.206
100	1.206									

Sumber: J NEMEC/Engineering hydrology

Nilai  $Y_N$  pada rumus 2.6 didapat dengan menggunakan tabel berikut sesuai dengan jumlah seri tahun data yang diperoleh:

**Tabel 2.3.** Harga reduced mean ( $Y_N$ )

M	0	1	2	3	4	5	6	7	8	9
10	0.495	0.499	0.503	0.507	0.510	0.512	0.515	0.518	0.520	0.522
20	0.523	0.525	0.526	0.528	0.529	0.530	0.532	0.533	0.534	0.535
30	0.536	0.537	0.538	0.538	0.539	0.540	0.541	0.541	0.542	0.543
40	0.543	0.544	0.544	0.545	0.545	0.546	0.546	0.547	0.547	0.548
50	0.548	0.548	0.549	0.549	0.550	0.550	0.550	0.551	0.551	0.551
60	0.552	0.552	0.552	0.553	0.553	0.553	0.553	0.554	0.554	0.554
70	0.554	0.555	0.555	0.555	0.555	0.555	0.556	0.556	0.556	0.556
80	0.556	0.557	0.557	0.557	0.557	0.557	0.558	0.558	0.558	0.558
90	0.558	0.558	0.558	0.559	0.559	0.559	0.559	0.559	0.559	0.559
100	0.560									

Sumber: *J NEMEC/Engineering hydrology*

Nilai  $Y_T$  pada rumus 2.6 didapat dengan menggunakan tabel berikut sesuai dengan periode ulang disain yang diinginkan:

**Tabel 2.4.** Harga reduced Variated ( $Y_T$ )

Periode Ulang (tahun)	Reduced Variate ( $Y_T$ )
2	0.3665
5	1.4999
10	2.2502
15	2.6844
20	2.9700
25	3.1985
50	3.9019

Sumber: *J NEMEC/Engineering hydrology*

### 2.2.1.2. Analisis Intensitas Hujan

Intensitas hujan yang tinggi pada umumnya berlangsung dengan durasi pendek dan meliputi daerah yang tidak sangat luas. Hujan yang meliputi daerah luas, jarang sekali dengan intensitas tinggi, tetapi dapat berlangsung dengan durasi cukup panjang. Kombinasi dari intensitas hujan yang tinggi dengan durasi panjang jarang terjadi, tetapi apabila terjadi berarti sejumlah besar volume air bagaikan ditumpahkan dari langit (Sudjarwadi 1987). Sri Harto (1993) menyebutkan bahwa analisis IDF memerlukan analisis frekuensi dengan menggunakan seri data yang diperoleh dari rekaman data hujan. Jika tidak tersedia waktu untuk

mengamati besarnya intensitas hujan atau disebabkan oleh karena alatnya tidak ada, dapat ditempuh cara-cara empiris dengan mempergunakan rumus-rumus eksperimental seperti rumus Talbot, Mononobe, Sherman dan Ishigura (Suyono dan Takeda 1993).

Intensitas curah hujan dapat dihitung berdasarkan rumus Mononobe yang merupakan variasi beberapa rumus intensitas curah hujan, rumus ini baik digunakan untuk curah hujan jangka pendek, setiap waktu berdasarkan curah hujan harian, yaitu:

$$I = \frac{X_{Tr24}}{24} \left( \frac{24}{T_d} \right)^{2/3} \quad (2.7)$$

dimana:  $X_{Tr24}$  = curah hujan harian rencana (mm)

$T_d$  = waktu konsentrasi atau  $T_c$  yaitu waktu yang diperlukan oleh air hujan untuk mengalir dari lokasi terjauh lintasan Daerah Aliran Sungai (DAS) menuju outlet DAS (jam)

$I$  = intensitas hujan (mm/jam)

Intensitas ( $I$ ) diketahui dari lengkung IDF dengan waktu jujuh (duration =  $D$ ) tertentu. Waktu jujuh ( $D$ ) diasumsikan terjadi saat waktu puncak banjir ( $T_p$ ) terjadi, sehingga waktu jujuh ( $D$ ) sama dengan waktu konsentrasi ( $T_c$ ).

### 2.2.1.3. Analisis Pola Sebaran Hujan

Untuk dapat mengkuantifikasi debit limpasan permukaan yang terjadi dalam suatu DAS, kedalaman hujan pada wilayah tersebut harus diketahui terlebih dahulu. Namun dalam suatu wilayah DAS yang cukup luas, kedalaman hujan menjadi sangat beragam besarnya, sehingga selain kedalaman hujan, pola sebaran kedalaman hujan juga perlu diperhatikan. Untuk menghitung pola sebaran hujan pada suatu DAS dapat dilakukan beberapa cara, antara lain rata-rata aljabar, polygon thiessen, dan isohyet.

a. Rata-rata Arithmetik

Metode ini merupakan metode yang paling mudah dimana setiap stasiun dianggap mempunyai bobot yang sama. Metode ini dapat memberikan hasil yang dapat dipercaya jika pos pengamatannya ditempatkan secara merata didalam wilayah, dan hasil penakaran masing-masing pos pengamatan tidak menyimpang jauh dari nilai rata-rata seluruh pos pengamatan diseluruh wilayah pengamatan. Metode ini cocok digunakan untuk daerah yang datar dan memiliki pos pengamatan curah hujan yang rapat dan banyak.

Tinggi curah hujan rata-rata dapat dihitung dengan rumus :

$$d = \frac{d_1 + d_2 + \dots + d_n}{n} = \sum_{i=1}^n \frac{d_i}{n} \quad (2.8)$$

dimana : d = tinggi curah hujan rata-rata  
 $d_1, d_2, \dots, d_n$  = tinggi curah hujan pada pos pengamatan  
 n = banyaknya pos pengamatan

b. Poligon Thiessen

Metode ini didasarkan rata-rata timbang/terbobot. Masing-masing pos pengamatan mempunyai daerah pengaruh yang dibentuk dengan menggambarkan garis-garis sumbu tegak lurus terhadap garis penghubung diantara dua buah pos penghubung.

Metode ini digunakan pada daerah dengan distribusi pengamatan curan hujan yang tidak tersebar merata didalam wilayah pengamatan. Hasil analisa dengan metode ini lebih teliti apabila dibandingkan dengan cara rata-rata aritmatik, karena dalam menentukan curah hujan wilayah dengan metode ini akan diperhitungkan persentase luas pengaruh masing-masing pos pengamatan curah hujan. Luas pengaruh pos pengamatan yang digunakan adalah luas daerah yang berada di dalam daerah aliran sungai.

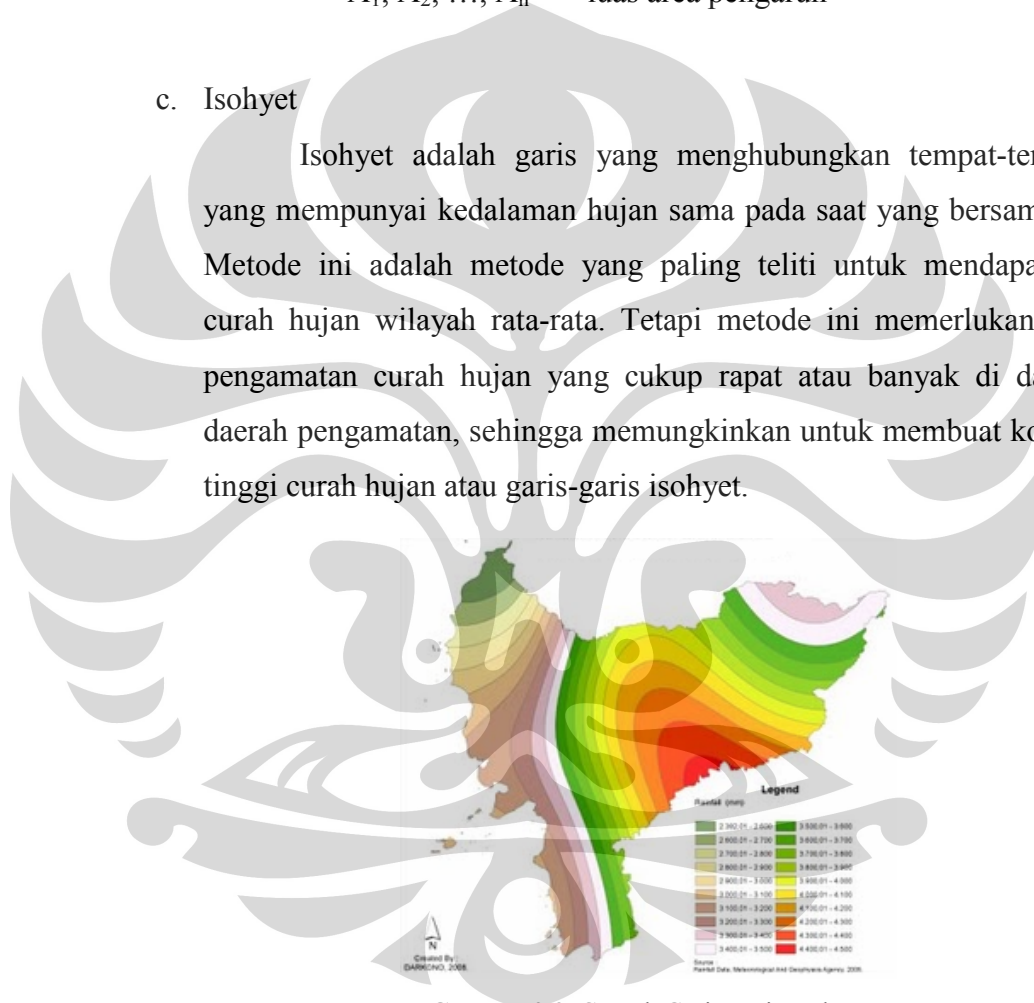
Rumus yang digunakan :

$$d = \frac{A_1 d_1 + A_2 d_2 + \dots + A_n d_n}{A_1 + A_2 + \dots + A_n} = \sum_{i=1}^n \frac{A_i d_i}{A_i} \quad (2.9)$$

dimana :  $A_t$  = luas area total  
 $d$  = tinggi curah hujan rata-rata  
 $d_1, d_2, \dots, d_n$  = tinggi curah hujan di pos pengamatan  
 $A_1, A_2, \dots, A_n$  = luas area pengaruh

### c. Isohyet

Isohyet adalah garis yang menghubungkan tempat-tempat yang mempunyai kedalaman hujan sama pada saat yang bersamaan. Metode ini adalah metode yang paling teliti untuk mendapatkan curah hujan wilayah rata-rata. Tetapi metode ini memerlukan pos pengamatan curah hujan yang cukup rapat atau banyak di dalam daerah pengamatan, sehingga memungkinkan untuk membuat kontur tinggi curah hujan atau garis-garis isohyet.



**Gambar 2.3.** Contoh Garis-garis Isohyet

Untuk mendapatkan suatu hasil penggambaran garis-garis isohyet yang diharapkan maka perlu diperhatikan kondisi topografi daerah seperti pengaruh bukit atau lembah. Rumus dari metode ini :

$$d = \frac{A_1 \frac{d_0 + d_1}{2} + A_2 \frac{d_1 + d_2}{2} + \dots + A_n \frac{d_{n-1} + d_n}{2}}{A_1 + A_2 + \dots + A_n} \quad (2.10)$$

dimana :  $d$  = tinggi curah hujan rata-rata  
 $d_0, d_1, \dots, d_n$  = curah hujan pada isohyet 0, 1, ..., n  
 $A_1, A_2, \dots, A_n$  = luas daerah yang dibatasi oleh isohyet yang bersangkutan.

### 2.2.2. Infiltrasi

Peresapan (infiltrasi) adalah pergerakan air menembus permukaan tanah dan masuk ke dalam tanah. Infiltrasi merupakan proses yang sangat kompleks, dan tergantung dari banyak faktor, seperti fisik tanah, keadaan tanah, keadaan penutup permukaan tanah (*vegetal cover*), serta intensitas dan lama hujan. Namun dari berbagai faktor tersebut, faktor yang paling menentukan adalah ukuran pori dan distribusinya.

Bila sebagian besar dari ruang pori-pori telah terisi dengan air, kapasitas peresapan biasanya menjadi lebih kecil jika dibandingkan saat tanah masih dalam kondisi yang kering. Bila ruang pori-pori tanah permukaan telah sepenuhnya terisi oleh air, maka gerakan air lebih ke bawah lagi akan tergantung pada permeabilitas tanah pada bagian bawahnya.

Infiltrasi mempunyai arti penting terhadap proses Limpasan. Daya infiltrasi menentukan besarnya air hujan yang dapat diserap ke dalam tanah. Sekali air hujan tersebut masuk ke dalam tanah ia akan diuapkan kembali atau mengalir sebagai air tanah. Makin besar daya infiltrasi, maka perbedaan antara intensitas curah dengan daya infiltrasi menjadi makin kecil. Akibatnya limpasan permukaannya makin kecil sehingga debit puncaknya juga akan lebih kecil.

Faktor-faktor yang mempengaruhi infiltrasi adalah:

1. Karakteristik-karakteristik hujan
2. Kondisi-kondisi permukaan tanah
  - Tetesan hujan, hewan maupun mesin mungkin memadatkan permukaan tanah dan mengurangi infiltrasi.
  - Pencucian partikel yang halus dapat menyumbat pori-pori pada permukaan tanah dan mengurangi laju infiltrasi.

- Laju infiltrasi awal dapat ditingkatkan dengan jeluk detensi permukaan.
  - Kepastian infiltrasi ditingkatkan dengan celah matahari.
  - Kemiringan tanah secara tidak langsung mempengaruhi laju infiltrasi selama tahapan awal hujan berikutnya.
  - Penggolongan tanah (dengan terasering, pembajakan kontur dll) dapat meningkatkan kapasitas infiltrasi karena kenaikan atau penurunan cadangan permukaan.
3. Kondisi-kondisi penutup permukaan
- Dengan melindungi tanah dari dampak tetesan hujan dan dengan melindungi pori-pori tanah dari penyumbatan, maka kemudian akan mendorong laju infiltrasi yang tinggi.
  - Salju mempengaruhi infiltrasi dengan cara yang sama seperti yang dilakukan butiran-butiran tanah yang menyumbat pori-pori.
  - Urbanisasi (bangunan, jalan, sistem drainase bawah permukaan) mengurangi infiltrasi.
4. Transmibilitas tanah
- Banyaknya pori yang besar, yang menentukan sebagian dari struktur tanah, merupakan salah satu faktor penting yang mengatur laju transmisi air yang turun melalui tanah.
  - Infiltrasi beragam secara terbalik dengan lengas tanah.
5. Karakteristik-karakteristik air yang berinfiltrasi
- Suhu air mempunyai banyak pengaruh, tetapi penyebabnya dan sifatnya belum pasti.
  - Kualitas air merupakan faktor lain yang mempengaruhi infiltrasi.

Salah satu metode yang dapat digunakan untuk mengkuantifikasi untuk mengkuantifikasi Infiltrasi pada suatu tanah, dapat dihitung dengan menggunakan persamaan empiris dari Horton (1939, dalam Schwab, G.O, and R.K. Frevert, 1981) yang mendefinisikan total volume dari infiltrasi setelah beberapa tenggang waktu dapat dihitung dengan menggunakan rumus berikut:

$$F = f_c t + \frac{(f_0 - f_c)}{k} \times (1 - e^{-kt}) \quad (2.11)$$

dimana:  $F$  = infiltrasi kumulatif pada  $t$  (mm/detik)

$f_c$  = laju infiltrasi pada saat infiltrasi telah konstan (mm/detik)

$f_0$  = laju infiltrasi awal (mm/detik)

$t$  = waktu (detik)

$k$  = konstanta (detik<sup>-1</sup>).

Dengan demikian maka besarnya laju infiltrasi pada suatu media tanah dapat diketahui dengan menurunkan persamaan tersebut, sehingga terbentuklah persamaan laju infiltrasi sebagai berikut:

$$f = f_c + (f_0 - f_c)e^{-kt} \quad (2.12)$$

dimana:  $f$  = laju infiltrasi pada  $t$  (mm/detik)

$f_c$  = laju infiltrasi pada saat infiltrasi telah konstan (mm/detik)

$f_0$  = laju infiltrasi awal (mm/detik)

$t$  = waktu (detik)

$k$  = konstanta (detik<sup>-1</sup>).

Besarnya koefisien laju infiltrasi awal ( $f_0$ ), laju infiltrasi pada saat infiltrasi telah konstan ( $f_c$ ), dan konstanta  $k$  ditentukan dengan menggunakan tabel berikut:

**Tabel 2.5.** Harga koefisien  $f_0$ ,  $f_c$ , dan  $k$

Soil Type	$f_0$ (mm/s)	$f_c$ (mm/s)	$k$ (s <sup>-1</sup> )
<b>Sandy soils</b>			
little or no vegetation	0.0353	0.0032	0.0055
dense vegetation	0.0706	0.0036	0.0052
poor developed	0.0172	0.0027	0.0050
well developed	0.0233	0.0030	0.0055
<b>Loam soils</b>			
little or no vegetation	0.0212	0.0008	0.0107
dense vegetation	0.0422	0.0011	0.0097
poor developed	0.0117	0.0004	0.0083
well developed	0.0171	0.0007	0.0107
<b>Clay soils</b>			
little or no vegetation	0.0071	0.0006	0.0022
dense vegetation	0.0142	0.0008	0.0157
poor developed	0.0056	0.0002	0.0017
well developed	0.0098	0.0005	0.0022

Sumber: Rangeland Watershed Program



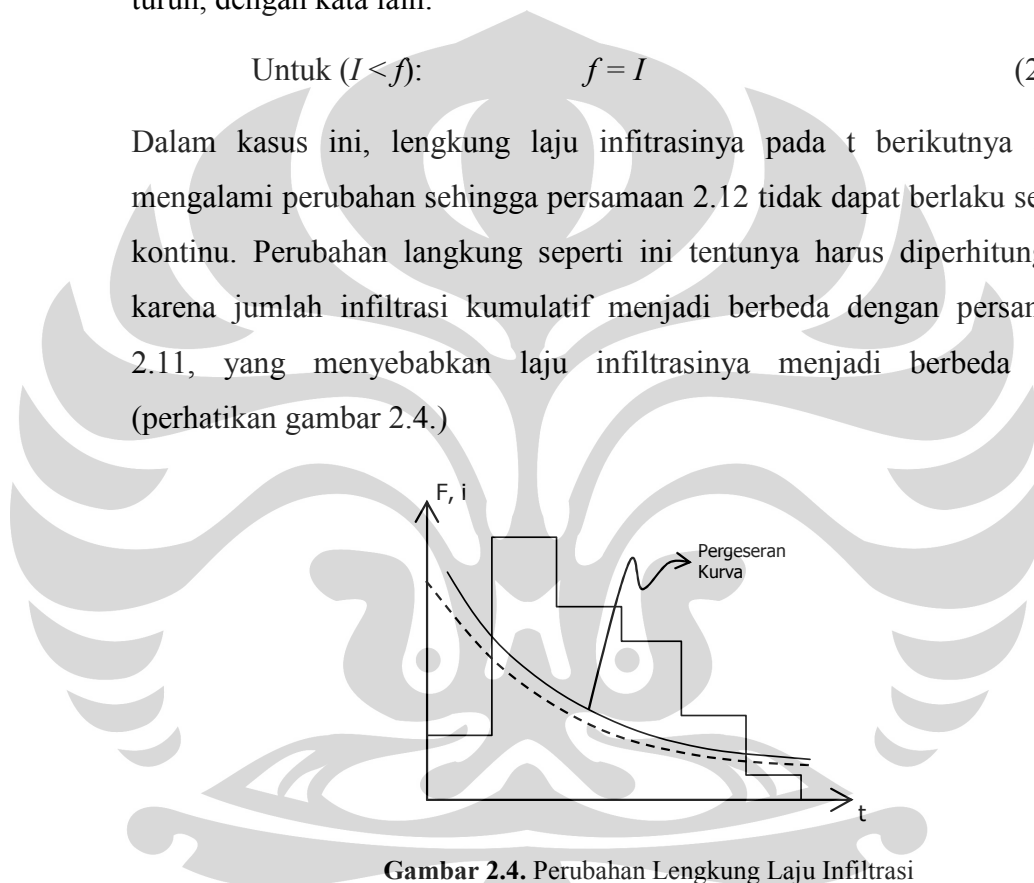
Persamaan laju infiltrasi diatas hanya berlaku apabila intensitas hujan lebih besar atau sama dengan laju infiltrasi ( $I \geq f$ ) pada saat bersamaan, dengan kata lain:

$$\text{Untuk } (I \geq f): \quad f = f_{\text{capacity}} \quad (2.13)$$

Sedangkan untuk intensitas hujan yang lebih kecil dari laju infiltrasi ( $I < f$ ), maka besarnya laju infiltrasi akan sama dengan intensitas hujan yang turun, dengan kata lain:

$$\text{Untuk } (I < f): \quad f = I \quad (2.14)$$

Dalam kasus ini, lengkung laju infiltrasinya pada  $t$  berikutnya akan mengalami perubahan sehingga persamaan 2.12 tidak dapat berlaku secara kontinu. Perubahan langkung seperti ini tentunya harus diperhitungkan karena jumlah infiltrasi kumulatif menjadi berbeda dengan persamaan 2.11, yang menyebabkan laju infiltrasinya menjadi berbeda pula (perhatikan gambar 2.4.)



**Gambar 2.4.** Perubahan Lengkung Laju Infiltrasi

Dengan demikian jika intensitas hujan yang lebih kecil dari laju infiltrasi maka terjadi pergeseran waktu  $t$  (menjadi lebih lambat) dimana pergeseran waktu  $t$  ini dapat dihitung dengan menggunakan rumus berikut:

$$t = -\frac{1}{k} \ln \left( \frac{f - f_c}{f_0 - f_c} \right) \quad (2.15)$$

dimana:  $f$  = laju infiltrasi pada kumulatif (mm/detik)

$f_c$  = laju infiltrasi pada saat infiltrasi telah konstan (mm/detik)

$f_0$  = laju infiltrasi awal (mm/detik)

$t$  = waktu (detik)

$k$  = konstanta ( $\text{detik}^{-1}$ ).

Untuk mempermudah perhitungan, maka rumus pergeseran  $t$  ini dipakai pada dua kondisi yaitu  $I \geq f$  dan  $I < f$ . Dimana besarnya nilai  $f$  untuk kondisi  $I \geq f$  dihitung dengan menggunakan rumus 2.12 dan besarnya nilai  $f$  pada kondisi  $I < f$  sama dengan intensitas hujan ( $I$ ). Untuk penjelasan dan penggunaan lebih lanjut rumusan ini akan dijelaskan pada bab selanjutnya.

### 2.3. Kecepatan Aliran Limpasan Permukaan

Dalam mengkuantifikasi aliran limpasan permukaan, tentunya tidak hanya cukup dengan menghitung kumulatif volume yang terjadi, karena masih banyak peristiwa fisik lain yang cukup kompleks yang menyebabkan besarnya aliran limpasan permukaan pada tiap wilayah menjadi sangat bervariasi. Salah satu dari peristiwa fisik tersebut adalah adanya pengaliran atau pergerakan limpasan permukaan secara gravitasi. Limpasan hujan tentunya tidak hanya akan menggenangi suatu wilayah, namun secara alamiah akan mengalir secara gravitasi menuju tempat yang lebih rendah dari posisi awal atau posisi sebelumnya. Besarnya kecepatan aliran limpasan pada permukaan tanah dapat dihitung dengan menggunakan rumus manning (*Irish engineer*, Robert Manning, 1890) sebagai berikut:

$$v = \frac{1}{n} R^{2/3} S^{1/2} \quad (2.16)$$

dimana:  $v$  = kecepatan rata-rata penampang (m/detik)

$n$  = koefisien kekasaran manning

$R$  = jari-jari hidrolis (m)

$S$  = kemiringan dasar (m/m).

Jari-jari hidrolis ( $R$ ) dapat dihitung dengan menggunakan rumus berikut:

$$R = \frac{A}{P} \quad (2.17)$$

dimana:  $R$  = jari-jari hirdolis (m)

A = luas penampang ( $m^2$ )

P = keliling basah (m).

Setelah dapat mengkuantifikasi besarnya limpasan permukaan dan dapat menghitung besarnya kecepatan aliran limpasan permukaan yang terjadi, maka kemudian dapat dilakukan simulasi atau pemodelan dengan cara membagi-bagi wilayah menjadi beberapa bagian yang sangat kecil. Dengan cara membagi wilayah seperti ini, maka akan meningkatkan presisi perhitungan dimana nantinya antar bagian akan memiliki keterkaitan satu sama lain terkait masalah volume limpasan permukaan dan kecepatannya.

Proses perhitungan seperti ini tentunya akan memakan waktu yang sangat lama apabila dilakukan secara manual, oleh karena itu akan digunakan alat bantu (*tools*) berupa Sistem Informasi Geografis, sehingga perhitungan semacam ini mungkin untuk dilakukan. Penjelasan lebih lanjut mengenai proses simulasi hujan-limpasan akan dijelaskan lebih detail pada bab berikutnya.

**BAB III**  
**APLIKASI SISTEM INFORMASI GEOGRAFIS**  
**DALAM KUANTIFIKASI LIMPASAN**

**3.1. Sistem Informasi Geografis (SIG)**

Sistem Informasi Geografis (SIG) merupakan sebuah sistem manajemen informasi yang menyeluruh, di dalamnya termasuk kegiatan survai, pemetaan, kartografi, fotogrametri, penginderaan jarak jauh dan ilmu komputer. Sistem ini memungkinkan pengguna untuk memasukkan data, mengatur, menganalisis, memanipulasi dan menampilkan data spasial. SIG mempunyai kemampuan untuk melakukan penyelidikan spasial dan *overlay* sehingga bisa menghasilkan informasi baru. (Glenn O. Schwab, 1996).

SIG dibagi menjadi dua kelompok yaitu sistem manual (analog), dan sistem otomatis (yang berbasis digital komputer). Perbedaan yang paling mendasar terletak pada cara pengelolaannya. Sistem informasi manual biasanya menggabungkan beberapa data seperti peta, lembar transparansi untuk tumpang susun (*overlay*), foto udara, laporan statistik dan laporan survey lapangan. Kesemua data tersebut dikompilasi dan dianalisis secara manual dengan alat tanpa komputer. Sedangkan SIG otomatis telah menggunakan komputer sebagai sistem pengolah data melalui proses digitasi. Sumber data digital dapat berupa citra satelit atau foto udara digital serta foto udara yang terdigitasi. Data lain dapat berupa peta dasar terdigitasi (Nurshanti, 1995).

Pengertian SIG saat ini lebih sering mengacu pada teknologi informasi spasial atau geografi yang berorientasi pada penggunaan teknologi komputer. Manajemen sistem informasi yang lengkap dengan menggunakan teknologi komputer menjadikan SIG sebagai sebuah alat bantu (*tools*) yang cukup populer dan telah banyak dipergunakan oleh berbagai kalangan untuk memecahkan berbagai permasalahan. Keterkaitan pemerintah dan berbagai organisasi dalam menyediakan sumber data menjadikan alat bantu ini semakin banyak diminati oleh kalangan *engineering*.

### 3.2. Komponen Utama SIG

Berdasarkan perangkat yang bekerja di dalamnya, SIG dibagi menjadi empat komponen utama:

#### 1. perangkat keras

SIG membutuhkan komputer untuk penyimpanan dan pemrosesan data. Ukuran dari sistem komputerisasi bergantung pada tipe SIG itu sendiri. SIG dengan skala kecil biasanya hanya membutuhkan sebuah PC (*personal computer*). Namun ketika suatu SIG dibuat dalam skala yang cukup besar, maka di perlukan spesifikasi komputer yang besar serta *host* untuk *client machine* yang mendukung penggunaan *multiple user*. Kebutuhan spesifikasi yang cukup besar tersebut lebih disebabkan karena data yang digunakan dalam SIG, baik data vektor maupun data raster, dalam penyimpanannya membutuhkan ruang yang cukup besar dan dalam tiap proses analisisnya membutuhkan memori yang besar serta prosesor yang cepat.

#### 2. perangkat lunak

Dalam pembuatan SIG di perlukan software yang menyediakan fungsi *tools* yang mampu melakukan penyimpanan data, analisis dan menampilkan informasi geografis. Dengan demikian, elemen yang harus terdapat dalam komponen software GIS adalah:

- *tool* untuk melakukan input dan transformasi data geografis;
- Sistem Manajemen Basis Data (DBMS);
- *tools* yang mendukung *query* geografis, analisa dan visualisasi; dan
- *Graphical User Interface* (GUI) untuk memudahkan akses pada *tools* geografi.

Inti dari software GIS adalah software GIS itu sendiri yang mampu menyediakan fungsi-fungsi untuk penyimpanan, pengaturan, *link*, *query* dan analisa data geografi. Beberapa contoh software GIS adalah ArcView, MapInfo, ArcInfo untuk SIG; CAD system untuk *entry graphic data*; dan ERDAS serta ER-MAP untuk proses *remote sensing data*. Modul dasar perangkat lunak SIG: modul pemasukan dan

pembetulan data, modul penyimpanan dan pengorganisasian data, modul pemrosesan dan penyajian data, modul transformasi data, modul interaksi dengan pengguna (*input query*).

3. organisasi atau manajemen
4. pemakai user.

Sedangkan Lukman (1993) menyatakan bahwa SIG terkait dengan penyajian informasi keruangan beserta atributnya, terdiri dari empat komponen utama yaitu:

1. Masukan data

Merupakan proses pemasukan data pada komputer dari peta (peta topografi dan peta tematik), data statistik, data hasil pengolahan citra digital penginderaan jauh, dan lain-lain. Data-data spasial dan atribut baik dalam bentuk analog maupun data digital tersebut dikonversikan kedalam format yang diminta oleh perangkat lunak sehingga terbentuklah sebuah basisdata (*database*).

2. Penyimpanan data dan pemanggilan kembali

Penyimpanan data dan pemanggilan kembali (*data storage and retrieval*) ialah penyimpanan data pada komputer dan pemanggilan kembali dengan cepat (penampilan pada layar monitor dan dapat ditampilkan/cetak pada kertas).

3. Manipulasi data dan analisis

Manipulasi data dan analisis ialah suatu kegiatan yang dapat dilakukan dengan berbagai macam perintah yang telah tersedia atau ditambahkan dalam suatu perangkat SIG, misalnya *overlay* antara dua tema peta, membuat *buffer zone* jarak tertentu dari suatu area atau titik dan sebagainya. Anon (2003) mengatakan bahwa manipulasi dan analisis data merupakan ciri utama dari SIG. Kemampuan SIG dalam melakukan analisis gabungan dari data spasial dan data atribut akan menghasilkan informasi yang berguna untuk berbagai aplikasi.

#### 4. Pelaporan data

Pelaporan data ialah suatu bentuk penyajian data dasar dan data hasil pengolahan dari suatu model komputer menjadi bentuk peta atau data tabular. Menurut Barus dan wiradisastra (2000) Bentuk produk suatu SIG dapat bervariasi baik dalam hal kualitas, keakuratan dan kemudahan pemakainya. Hasil ini dapat dibuat dalam bentuk peta-peta, tabel angka-angka, teks di atas kertas atau media lain (*hard copy*), atau dalam cetak lunak (seperti file elektronik).

### 3.3. Kemampuan Dasar SIG

Kemampuan dasar SIG yang paling banyak dipakai oleh berbagai kalangan di seluruh dunia diantaranya adalah kemampuan dalam memetakan letak, kemampuan dalam memetakan kuantitas, kemampuan dalam memetakan kerapatan, dan kemampuan dalam memetakan perubahan. Berbagai kemampuan inilah yang menjadikan SIG menjadi salah satu perangkat yang paling digemari sebagai perangkat manajemen dan pengolahan data. Dari beberapa kemampuan dasar tersebut, tentunya dapat dikembangkan lebih lanjut dengan kemampuan tambahan lain sesuai dengan kebutuhan penggunaannya. Beberapa kemampuan dasar SIG akan dibahas lebih lanjut dalam sub-bab berikutnya.

#### 3.3.1. Memetakan Letak

Data sesungguhnya di permukaan bumi akan dipetakan ke dalam beberapa layer dengan setiap layer-nya merupakan representasi kumpulan *feature* yang mempunyai kesamaan, contohnya *layer* jalan, *layer* bangunan, dan sebagainya. *Layer* demi *layer* ini kemudian disatukan dengan disesuaikan urutannya. Setiap data pada setiap *layer* dapat dicari, seperti halnya melakukan *query* terhadap *database*, untuk kemudian dilihat letaknya dalam keseluruhan peta. Kemampuan ini memungkinkan seseorang untuk mencari dimana letak suatu daerah, benda, atau lainnya di permukaan bumi. Fungsi ini dapat digunakan seperti untuk mencari lokasi rumah, mencari rute jalan, mencari tempat-tempat penting dan lainnya

yang ada di peta. Orang dapat pula melihat pola-pola yang mungkin akan muncul dengan melihat penyebaran letak-letak *feature*, misalnya daerah rawan banjir, daerah perumahan, daerah hutan dan sebagainya.

### 3.3.2. Memetakan Kuantitas

Orang sering memetakan kuantitas, yaitu sesuatu yang berhubungan dengan jumlah, seperti dimana yang paling banyak atau dimana yang paling sedikit. Dengan melihat penyebaran kuantitas tersebut dapat mencari tempat-tempat yang sesuai dengan kriteria yang diinginkan dan digunakan untuk pengambilan keputusan, ataupun juga untuk mencari hubungan dari masing-masing tempat tersebut. Pemetaan ini akan lebih memudahkan pengamatan terhadap data statistik dibanding *database* biasa.

### 3.3.3. Memetakan Kerapatan

Sewaktu orang melihat konsentrasi dari penyebaran lokasi dari *feature*, di wilayah yang mengandung banyak *feature* mungkin akan mendapat kesulitan untuk melihat wilayah mana yang mempunyai konsentrasi lebih tinggi dari wilayah lainnya. Peta kerapatan dapat mengubah bentuk konsentrasi kedalam unit-unit yang lebih mudah untuk dipahami dan seragam, misal membagi dalam kotak-kotak selebar 10 km<sup>2</sup>, dengan menggunakan perbedaan warna untuk menandai tiap-tiap kelas kerapatan.

### 3.3.4. Memetakan Perubahan

Dengan memasukkan variabel waktu, SIG dapat dibuat untuk peta historikal. Histori ini dapat digunakan untuk memprediksi keadaan yang akan datang dan dapat pula digunakan untuk evaluasi kebijaksanaan. Seperti contoh pemetaan jalur yang dilalui banjir, dapat digunakan untuk memprediksi kemana nantinya arah evakuasi atau wilayah yang harus waspada terhadap serangan banjir dan sebagainya.



### 3.4. Sumber Data dalam SIG

Untuk melakukan analisis dengan menggunakan SIG, tentunya diperlukan berbagai sumber data yang terkait dengan analisis yang akan dilakukan. Berbagai data tersebut umumnya berasal dari sumber yang telah tersedia sebelumnya, baik yang telah disediakan oleh pemerintah maupun data yang berasal dari perangkat informasi lain seperti citra satelit maupun *Global Positioning System* (GPS). Keseluruhan data tersebut tentunya harus dilakukan proses digitasi lebih lanjut agar data tersebut dapat digunakan dalam melakukan analisis dengan menggunakan SIG. Beberapa sumber data yang paling umum digunakan pada SIG antara lain berupa peta analog, penginderaan jauh, pengukuran lapangan, dan GPS.

#### 3.4.1. Peta Analog

Peta analog merupakan peta yang disediakan dalam bentuk cetak seperti peta topografi, peta tanah, peta rupa bumi, dan sebagainya. Pada umumnya peta analog dibuat dengan teknik kartografi sehingga berbagai referensi spasial telah disediakan secara lengkap seperti: koordinat, skala, dan arah mata angin. Hal ini tentu saja dapat mempermudah proses pemindahan data ini ke dalam SIG, sehingga peta analog dapat langsung dikonversi menjadi peta digital dan siap digunakan pada SIG.

#### 3.4.2. Citra Satelit

Data ini menggunakan satelit sebagai wahananya. Satelit tersebut menggunakan sensor untuk dapat merekam kondisi atau gambaran dari permukaan bumi. Umumnya diaplikasikan dalam kegiatan yang berhubungan dengan pemantauan sumber daya alam di permukaan bumi (bahkan ada beberapa satelit yang sanggup merekam hingga dibawah permukaan bumi), studi perubahan lahan dan lingkungan, dan aplikasi lain yang melibatkan aktifitas manusia di permukaan bumi. Kelebihan dari teknologi terutama dalam dekade ini adalah dalam kemampuan merekam cakupan wilayah yang luas dan tingkat resolusi dalam merekam obyek yang sangat tinggi. Data yang dihasilkan dari citra satelit kemudian

diturunkan menjadi data tematik dan disimpan dalam bentuk basis data untuk digunakan dalam berbagai macam aplikasi.

### 3.4.3. Pengukuran Lapangan

Data pengukuran lapangan, yaitu data yang dihasilkan berdasarkan teknik perhitungan tersendiri, pada umumnya data ini merupakan sumber data atribut, beberapa bentuk pengukuran lapangan yang umum adalah data tabular dan data survei.

- Data Tabular, data ini berfungsi sebagai atribut bagi data spasial dan umumnya berbentuk tabel. Salah satu contoh data ini yang paling banyak digunakan adalah data sensus penduduk, data sosial, data ekonomi, dll. Data tabular ini kemudian di relasikan dengan data spasial untuk menghasilkan tema data tertentu.
- Data Survei (Pengamatan atau pengukuran dilapangan), data ini dihasilkan dari hasil survei atau pengamatan dilapangan. Contohnya adalah pengukuran persil lahan dengan menggunakan metode survei terestris.

### 3.4.4. Global Positioning Sistem (GPS)

Teknologi GPS memberikan terobosan penting dalam menyediakan data bagi SIG. Keakuratan pengukuran GPS semakin tinggi dengan berkembangnya teknologi. Data ini biasanya direpresentasikan dalam format vektor.

### 3.4.5. Foto Udara

Foto Udara (*Aerial Photographs*) merupakan salah satu sumber data yang banyak digunakan untuk menghasilkan data spasial selain dari citra satelit. Perbedaannya dengan citra satelit hanya terletak pada jenis wahana yang digunakan dan cakupan wilayahnya. Biasanya foto udara menggunakan pesawat udara. Secara teknis proses pengambilan atau

perekaman datanya hampir sama dengan citra satelit.

Sebelum berkembangnya teknologi kamera digital, kamera yang digunakan adalah menggunakan kamera konvensional menggunakan negatif film, saat ini sudah menggunakan kamera digital, dimana data hasil perekaman dapat langsung disimpan dalam basis data. Sedangkan untuk data lama (format foto film) agar dapat disimpan dalam basis data harus dilakukan konversi dahulu dengan menggunakan *scanner*, sehingga dihasilkan foto udara dalam format digital.

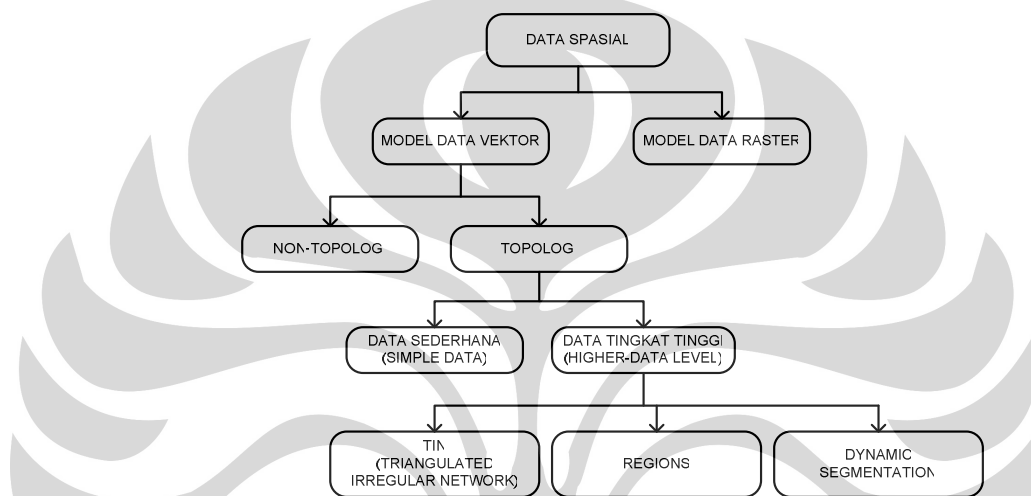
### 3.5. Data Spasial dalam SIG

Data spasial mempunyai pengertian sebagai suatu data yang mengacu pada posisi, obyek, dan hubungan diantaranya dalam ruang bumi. Data spasial merupakan salah satu item dari informasi, dimana didalamnya terdapat informasi mengenai bumi termasuk permukaan bumi, dibawah permukaan bumi, perairan, kelautan dan bawah atmosfer (Rajabidfard dan Williamson, 2000a). Data spasial dan informasi turunannya digunakan untuk menentukan posisi dari identifikasi suatu elemen di permukaan bumi (Radjabidfard 2001). Lebih lanjut lagi *Mapping Science Committee* (1995) dalam Rajabidfard (2001) menerangkan mengenai pentingnya peranan posisi lokasi yaitu:

- pengetahuan mengenai lokasi dari suatu aktifitas memungkinkan hubungannya dengan aktifitas lain atau elemen lain dalam daerah yang sama atau lokasi yang berdekatan dan
- Lokasi memungkinkan diperhitungkannya jarak, pembuatan peta, memberikan arahan dalam membuat keputusan spasial yang bersifat kompleks.

Pada pemanfaatannya, data spasial yang diolah dengan menggunakan komputer (data spasial digital) menggunakan model sebagai pendekatannya. *Economic and Social Commission for Asia and the Pacific* (1996), mendefinisikan model data sebagai suatu set logika atau aturan dan karakteristik dari suatu data spasial. Model data merupakan representasi hubungan antara dunia nyata dengan data spasial.

Terdapat dua model dalam data spasial, yaitu *model data raster* dan *model data vektor*. Keduanya memiliki karakteristik yang berbeda, selain itu dalam pemanfaatannya tergantung dari masukan data dan hasil akhir yang akan dihasilkan. Model data tersebut merupakan representasi dari obyek-obyek geografi yang terekam sehingga dapat dikenali dan diproses oleh komputer. Chang (2002) menjabarkan model data vektor menjadi beberapa bagian lagi (dapat dilihat pada Gambar 3.1.), sedangkan penjelasan dari model data tersebut akan dibahas dalam sub bab berikut ini.



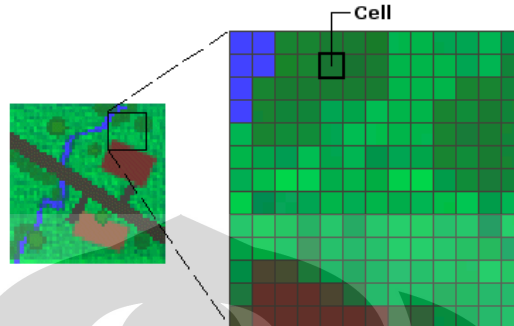
**Gambar 3.1.** Klasifikasi Model Data Spasial

### 3.5.1. Model Data Raster

Model data raster mempunyai struktur data yang tersusun dalam bentuk matriks atau piksel dan membentuk grid. Setiap piksel memiliki nilai tertentu dan memiliki atribut tersendiri, termasuk nilai koordinat yang unik. Tingkat keakuratan model ini sangat tergantung pada ukuran piksel atau biasa disebut dengan *resolusi*. Model data ini biasanya digunakan dalam *remote sensing* yang berbasis citra satelit maupun *airborne* (pesawat terbang). Selain itu model ini digunakan pula dalam membangun model ketinggian digital (*DEM-Digital Elevation Model*) dan model permukaan digital (*DTM-Digital Terrain Model*).

Model raster memberikan informasi spasial terhadap permukaan di bumi dalam bentuk gambaran yang di generalisasi. Representasi dunia nyata disajikan sebagai elemen matriks atau piksel yang membentuk grid

yang homogen. Pada setiap piksel mewakili setiap obyek yang terekam dan ditandai dengan nilai-nilai tertentu. Secara konseptual, model data raster merupakan model data spasial yang paling sederhana.



**Gambar 3.2.** Struktur Model Data Raster

Karakteristik utama data raster adalah bahwa dalam setiap sel/piksel mempunyai nilai. Nilai sel/piksel merepresentasikan fenomena atau gambaran dari suatu kategori. Nilai sel/piksel dapat memiliki nilai positif atau negatif, *integer*, dan *floating point* untuk dapat merepresentasikan nilai *continuous* (lihat Gambar 3.3). Data raster disimpan dalam suatu urutan nilai sel/piksel. Sebagai contoh, 80, 74, 45, 45, 34, dan seterusnya.

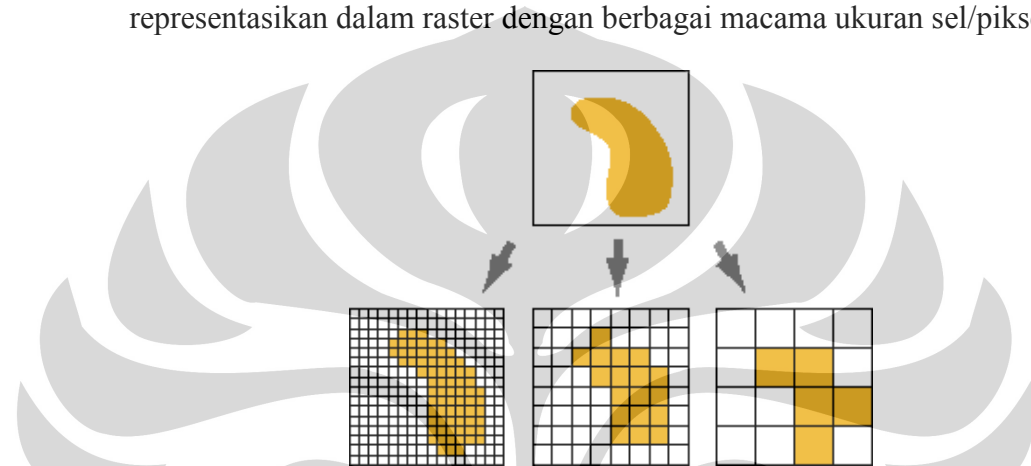
80	74	62	45	45	34	39	56
80	74	74	62	45	34	39	56
74	74	62	62	45	34	39	39
62	62	45	45	34	34	34	39
45	45	45	34	34	30	34	39

**Gambar 3.3** Struktur Penyimpanan Model Data Raster

Luas suatu area direpresentasikan dalam setiap sel/piksel dengan lebar dan panjang yang sama. Sebagai contoh, sebuah data raster yang merepresentasikan ketinggian permukaan (biasa disebut dengan DEM) dengan luasan sebesar 100 Km<sup>2</sup>, apabila terdapat 100 sel/piksel dalam raster, maka dalam setiap sel/piksel mempunyai ukuran 1 Km<sup>2</sup> (1 km x 1 km).

Dimensi dari setiap sel/piksel dapat ditentukan ukurannya sesuai dengan kebutuhan. Ukuran sel/piksel menentukan bagaimana kasar atau halus pola atau obyek yang akan di representasikan. Semakin kecil

ukuran sel/piksel, maka akan semakin halus atau lebih detail. Akan tetapi semakin besar jumlah sel/piksel yang digunakan maka akan berpengaruh terhadap penyimpanan dan kecepatan proses. Apabila ukuran sel /piksel terlalu besar akan terjadi kehilangan informasi atau kehalusan pola akan terlihat lebih kasar. Sebagai contoh apabila ukuran sel lebih besar dari lebar jalan, maka jalan tidak akan dapat ditampilkan dalam data raster. Gambar berikut memperlihatkan bagaimana obyek poligon di representasikan dalam raster dengan berbagai macam ukuran sel/piksel.



**Gambar 3.4.** Poligon dalam Berbagai Macam Ukuran Sel

Lokasi dalam setiap sel/piksel di definisikan dalam bentuk baris dan kolom dimana didalamnya terdapat informasi mengenai posisi. Apabila sel memuat Sistem Koordinat Kartesian, dimana setiap baris merupakan paralel dengan sumbu X (*x-axis*), dan kolom paralel dengan sumbu Y (*y-axis*). Demikian pula apabila sel/piksel memuat Sistem Koordinat UTM (*Universal Transverse Mercator*) dan sel/piksel memiliki ukuran 100, maka lokasi sel/piksel tersebut pada 300, 500 E (*east*) dan 5, 900, 600 N (*north*).

Terdapat beberapa keuntungan dalam menggunakan model raster, diantaranya adalah :

- Memiliki struktur data yang sederhana, bentuk sel matriks dengan nilainya dapat merepresentasikan koordinat dan kadangkala memiliki *link* dengan tabel atribut.
- Format yang sangat cocok untuk dapat melakukan analisis statistik dan spasial.

- Mempunyai kemampuan dalam merepresentasikan data-data yang bersifat *continous* seperti dalam memodelkan permukaan bumi.
- Memiliki kemampuan untuk menyimpan titik (*point*), garis (*line*), area (*polygon*), dan permukaan (*surface*)
- Memiliki kemampuan dalam melakukan proses tumpang-tindih (*overlay*) secara lebih cepat pada data yang kompleks.

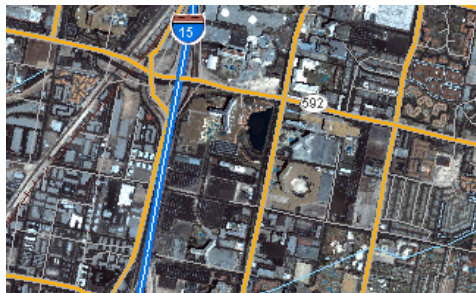
Selain keuntungan dari model raster, terdapat pula beberapa pertimbangan yang perlu diperhatikan dalam menggunakan model data raster dibandingkan dengan data vektor, diantaranya adalah :

- Terdapat beberapa keterbatasan masalah akurasi dan presisi data terutama dalam pada saat menentukan ukuran sel/piksel.
- Data raster sangat berpotensi dalam menghasilkan ukuran file yang sangat besar. Peningkatan resolusi akan meningkatkan ukuran data, hal ini akan berdampak pada penyimpanan data dan kecepatan proses. Hal ini akan sangat bergantung kepada kemampuan hardware yang akan digunakan.

Pemanfaatan model data raster banyak digunakan dalam berbagai aplikasi, akan tetapi Environmental Systems Research Institute (ESRI), Inc (2006) membagi menjadi empat kategori utama, yaitu :

- Raster sebagai peta dasar

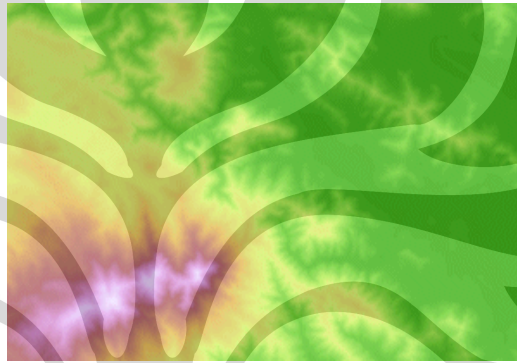
Data raster Biasanya digunakan sebagai tampilan latar belakang (*background*) untuk suatu layer dari obyek yang lain (vektor). Sebagai contoh foto udara ortho ditampilkan sebagai latar dari obyek jalan (lihat Gambar 3.5). Tiga sumber utama dari peta dasar raster adalah foto udara, citra satelit, dan peta hasil scan.



**Gambar 3.5.** Foto Udara (Raster) Sebagai Latar dari Layer Jalan (Vektor)

- Raster sebagai peta model permukaan

Data raster sangat cocok untuk merepresentasikan data permukaan bumi. Data dapat menyediakan metode yang efektif dalam menyimpan informasi nilai ketinggian yang diukur dari permukaan bumi. Selain dapat merepresentasikan permukaan bumi, data raster dapat pula merepresentasikan curah hujan, temperatur, konsentrasi, dan kepadatan populasi. Gambar 3.6 berikut ini memperlihatkan nilai ketinggian suatu permukaan bumi. Warna hijau memperlihatkan permukaan yang rendah, dan berikutnya merah, pink dan putih menunjukkan permukaan yang semakin tinggi.

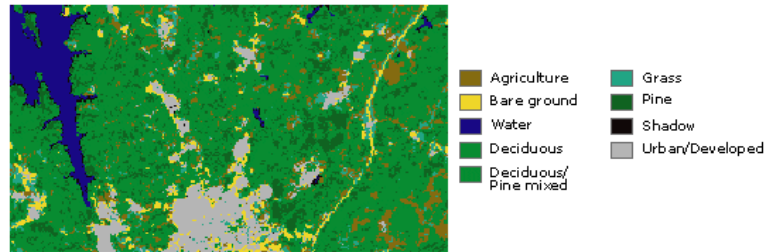


**Gambar 3.6.** Data Raster dalam Memodelkan Permukaan Bumi

- Raster sebagai peta tematik

Data raster yang merpresentasikan peta tematik dapat diturunkan dari hasil analisis data lain. Aplikasi analisis yang sering digunakan adalah dalam melakukan klasifikasi citra satelit untuk menghasilkan kategori tutupan lahan (*land cover*). Pada dasarnya aktifitas yang dilakukan adalah mengelompokkan nilai dari data multispektral kedalam kelas tertentu (seperti tipe vegetasi) dan memberikan nilai terhadap kategori tersebut. Peta tematik juga dapat dihasilkan dari operasi *geoprocessing* yang dikombinasikan dari berbagai macam sumber, seperti vektor, raster, dan data permukaan.





**Gambar 3.7.** Data Raster dalam Mengklasifikasi Data Tutupan Lahan

- Raster sebagai atribut dari obyek

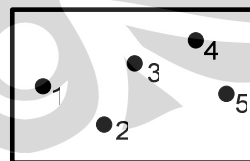
Data raster dapat pula digunakan sebagai atribut dari suatu obyek, baik dalam foto digital, dokumen hasil scan atau gambar hasil scan yang mempunyai hubungan dengan obyek geografi atau lokasi.

### 3.5.2. Model Data Vektor

Model data vektor merupakan model data yang paling banyak digunakan, model ini berbasiskan pada titik (points) dengan nilai koordinat (x,y) untuk membangun obyek spasialnya. Obyek yang dibangun terbagi menjadi tiga bagian lagi yaitu berupa titik (point), garis (line), dan area (polygon).

- Titik (point)

Titik merupakan representasi grafis yang paling sederhana pada suatu obyek. Titik tidak mempunyai dimensi tetapi dapat ditampilkan dalam bentuk simbol baik pada peta maupun dalam layar monitor. Contoh : Lokasi Fasilitas Kesehatan, Lokasi Fasilitas Kesehatan, dll.

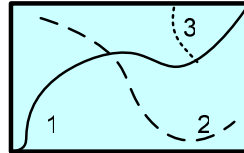


ID	Nama	Lokasi
1	SMU 1	Kec. A
2	SDN B	Kec. A
3	SMP 5	Kec. A
4	SDN A	Kec. B

**Gambar 3.8.** Representasi Data Titik

- Garis (line)

Garis merupakan bentuk linear yang menghubungkan dua atau lebih titik dan merepresentasikan obyek dalam satu dimensi. Contoh : Jalan, Sungai, dll.

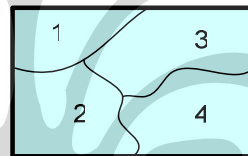


ID	Status Jalan	Kondisi
1	Jalan Nasional	Baik
2	Jalan Provinsi	Sedang
3	Jalan Kabupaten	Rusak

**Gambar 3.9.** Representasi Data Garis

- Area (Poligon)

Poligon merupakan representasi obyek dalam dua dimensi. Contoh : Danau, Persil Tanah, dll.



ID	Guna Lahan	Luas (Ha)
1	Sawah	20
2	Permukiman	30
3	Kebun	45
4	Danau	40

**Gambar 3.10.** Representasi Data Area

Seperti yang telah dijelaskan sebelumnya, model data vektor terbagi menjadi beberapa bagian, diantaranya:

- Topologi, biasa digunakan dalam analisis spasial dalam SIG. Topologi merupakan model data vektor yang menunjukkan hubungan spasial diantara obyek spasial. Topologi sangat berguna pada saat melakukan deteksi kesalahan pada saat proses digitasi. Selain itu berguna pula dalam melakukan proses analisis spasial yang bersifat kompleks dengan melibatkan data spasial yang cukup besar ukuran filenya. Salah satu contoh analisis spasial yang dapat dilakukan dalam format topologi adalah proses tumpang tindih (*overlay*) dan analisis jaringan

(*network analysis*) dalam SIG.

- Non Topologi, merupakan model data yang mempunyai sifat yang lebih cepat dalam menampilkan, dan yang paling penting dapat digunakan secara langsung dalam perangkat lunak (*software*) SIG yang berbeda-beda. Non-topologi digunakan dalam menampilkan atau memproses data spasial yang sederhana dan tidak terlalu besar ukuran filenya.
- Model data vektor dalam topologi lebih jauh lagi dapat dikembangkan dalam dua kategori, yaitu Data Sederhana (*Simple Data*) yang merupakan representasi data yang mengandung tiga jenis data (titik, garis, poligon) secara sederhana. Sedangkan Data Tingkat Tinggi (*Higher Data Level*), dikembangkan lebih jauh dalam melakukan pemodelan secara tiga dimensi (3 Dimensi/3D). Model tersebut adalah dengan menggunakan TIN (*Triangulated Irregular Network*). Model TIN merupakan suatu set data yang membentuk segitiga dari suatu data set yang tidak saling bertumpukan. Pada setiap segitiga dalam TIN terdiri dari titik dan garis yang saling terhubung sehingga membentuk segitiga. Model TIN berguna dalam merepresentasikan ruang (spasial) dalam bentuk 3D, sehingga dapat mendekati kenyataan di lapangan. Salah satu diantaranya adalah dalam membangun Model Permukaan Bumi Digital (*Digital Terrain Model / DTM*).
- Region, merupakan sekumpulan poligon, dimana masing-masing poligon tersebut dapat atau tidak mempunyai keterkaitan diantaranya akan tetapi saling bertumpukan dalam satu data set.
- Dynamic Segmentation, adalah model data yang dibangun dengan menggunakan segmen garis dalam rangka membangun model jaringan (*network*).

### 3.5.3. Perbandingan Model Data Raster dan Vektor

Kedua model data spasial yang telah disebutkan diatas (raster dan vektor) mempunyai karakteristik yang berbeda dalam mengaplikasikannya. Hal ini

sangat bergantung pada tujuan, analisis, sistem dan aplikasi yang akan digunakan. Tabel berikut ini memperlihatkan perbandingan diantara kedua model tersebut.

**Tabel 3.1.** Perbandingan Struktur Data Vektor dan Raster

<b>Parameter</b>	<b>Vektor</b>	<b>Raster</b>
Akurasi	Akurat dan lebih presisi	Sangat bergantung dengan ukuran grid/sel
Atribut	Relasi langsung dengan DBMS (database)	Grid/sel merepresentasikan atribut. Relasi dengan DBMS tidak secara langsung
Kompleksitas	Tinggi. Memerlukan algoritma dan proses yang sangat kompleks	Mudah dalam mengorganisasi dan proses
Output	Kualitas tinggi sangat bergantung dengan plotter/printer dan kartografi	Bergantung terhadap output printer/plotter
Analisis	Spasial dan atribut terintegrasi. Kompleksitasnya sangat tinggi	Bergantung dengan algoritma dan mudah untuk dianalisis
Simulasi	Kompleks dan sulit	Mudah untuk dilakukan simulasi
Input	Digitasi, dan memerlukan konversi dari scanner	Sangat memungkinkan untuk diaplikasikan dari hasil konversi dengan menggunakan scan
Volume	Bergantung pada kepadatan dan jumlah verteks	Bergantung pada ukuran grid/sel
Resolusi	Bermacam-macam	Tetap

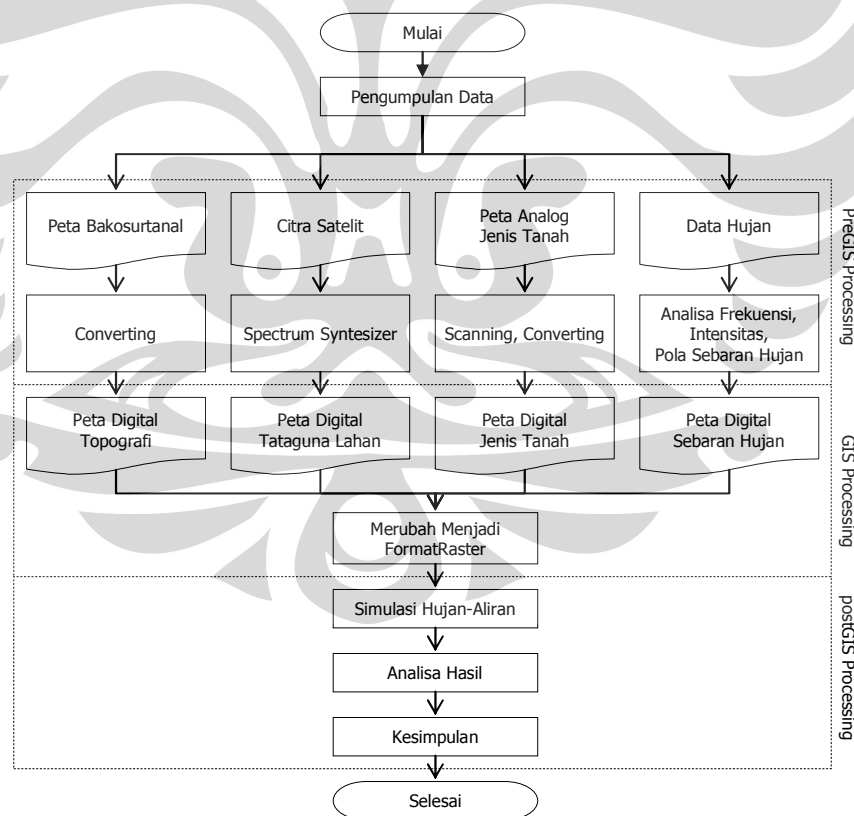
### 3.6. Kuantifikasi Limpasan Permukaan dengan menggunakan SIG

Seperti yang telah dijelaskan sebelumnya, sebuah perangkat SIG biasanya hanya dibekali dengan beberapa perintah yang umum dan banyak dipergunakan oleh penggunanya seperti *overlay*, *buffer*, *query*, dll. Namun terkadang untuk melengkapi suatu analisa yang cukup kompleks, sebuah perangkat SIG perlu dibekali dengan perangkat-perangkat tambahan (*tools extension*) yang sesuai dengan kebutuhan analisa.

Perangkat tambahan dalam SIG biasanya telah dikembangkan dan banyak tersedia di pasaran baik yang gratis maupun yang komersil. Namun sebagai *user*, kita juga dapat membuat dan mengembangkan sendiri perangkat tambahan yang sesuai dengan keinginan dan kebutuhan analisa yang kita inginkan. Hal ini tentu saja memungkinkan, karena kebanyakan

dari perangkat SIG merupakan perangkat dengan sistem arsitektur terbuka (*Open Architecture*) yang menyediakan berbagai API (*Application Programming Interface*) yang lengkap sehingga user dapat membuat perangkat tambahan tanpa harus tahu lebih jauh bagaimana perangkat SIG itu dibuat. Hal inilah yang akan dilakukan dalam kajian ini, yaitu membuat perangkat tambahan dari SIG yang berguna untuk mensimulasikan hujan-limpasan sehingga didapatkan kuantifikasi limpasan permukaan dan infiltrasi.

Sebelum data masuk kedalam perangkat tambahan yang akan dibuat, tentunya data tersebut harus melalui berbagai proses terlebih dahulu sehingga data masukkan menjadi sesuai dengan kriteria dan spesifikasi yang dibutuhkan oleh perangkat tambahan tersebut. Berbagai proses utama tersebut akan dibagi menjadi tiga proses utama, yaitu: *preGIS Processing*, *GIS Processing*, dan *postGIS Processing*.



**Gambar 3.11.** Proses Kuantifikasi Limpasan Permukaan dengan menggunakan SIG

### 3.6.1. *PreGIS Processing*

*preGIS Processing* adalah proses dimana data-data yang telah dikumpulkan dipersiapkan terlebih dahulu melalui proses *scanning*, *converting*, dan berbagai proses lainnya sampai data yang masih berupa data mentah (data tabulasi, peta manual, dll) dapat dibaca dan di proses lebih lanjut dengan menggunakan perangkat SIG. Pada kajian ini beberapa data mentah yang perlu dilakukan *preGIS Processing* adalah:

- Peta topografi yang berasal dari peta bakosurtanal masih berupa file CAD, sehingga diperlukan konversi menjadi file vektor (*shape file*) yang dapat didukung oleh perangkat SIG.
- Peta tataguna lahan diambil dari citra satelit sehingga datanya masih berupa *raw file*, untuk merubah data *raw* menjadi data atribut yang diinginkan (perumahan, persawahan, lapangan, dll) perlu dilakukan pemisahan spektrum warna sehingga didapat hasil akhir berupa peta raster dengan atribut yang sesuai.
- Peta jenis tanah masih berupa peta analog (*hard copy*) sehingga perlu dilakukan pemindaian dan digitasi menjadi peta raster sesuai dengan atribut yang sesuai (pasir, lanau, lempung, dll).
- Data hujan masih berupa data tabulasi tinggi hujan harian sehingga diperlukan perhitungan analisis frekuensi dan intensitas sehingga didapatkan laju intensitas yang sesuai dengan kejadian hujan yang mungkin terjadi. Pola sebaran intensitas hujan dilakukan dengan metode isohyet dengan tiga stasiun hujan, sehingga nantinya akan didapat hasil akhir berupa distribusi intensitas berupa file vektor (*shape file*) yang dapat dibaca oleh perangkat SIG.

### 3.6.2. *GIS Processing*

*GIS Processing* adalah proses dimana data-data yang telah siap dan dapat dibaca oleh perangkat SIG diolah terlebih dahulu sebelum masuk kedalam perangkat tambahan, sehingga data dapat terbaca dan simulasi dapat dilakukan. Data yang berasal dari SIG masih tercampur antara data vektor dan raster, kemudian data ini diseragamkan menjadi data yang

dapat dibaca oleh perangkat tambahan (*raw raster*) dengan ukuran dan resolusi yang sama ( $1\text{m}\times 1\text{m}$  *per pixel*). Pada kajian ini beberapa data olahan awal yang perlu dilakukan *GIS Processing* adalah:

- Peta topografi yang telah diolah menjadi file vektor, akan dilakukan konversi dengan menggunakan SIG menjadi data *raw raster* dengan resolusi  $1\text{m}\times 1\text{m}$  *per pixel*
- Peta tataguna lahan yang telah diolah menjadi file raster akan di sesuaikan resolusinya menjadi  $1\text{m}\times 1\text{m}$  *per pixel* dengan menggunakan perangkat SIG.
- Peta jenis tanah yang telah dipindai dan didigitasi akan dirubah menjadi file raster dan di sesuaikan resolusinya menjadi  $1\text{m}\times 1\text{m}$  *per pixel* dengan menggunakan perangkat SIG.
- Peta sebaran intensitas hujan yang telah diolah menjadi file vektor, akan dilakukan konversi dengan menggunakan SIG menjadi data *raw raster* dengan resolusi  $1\text{m}\times 1\text{m}$ .

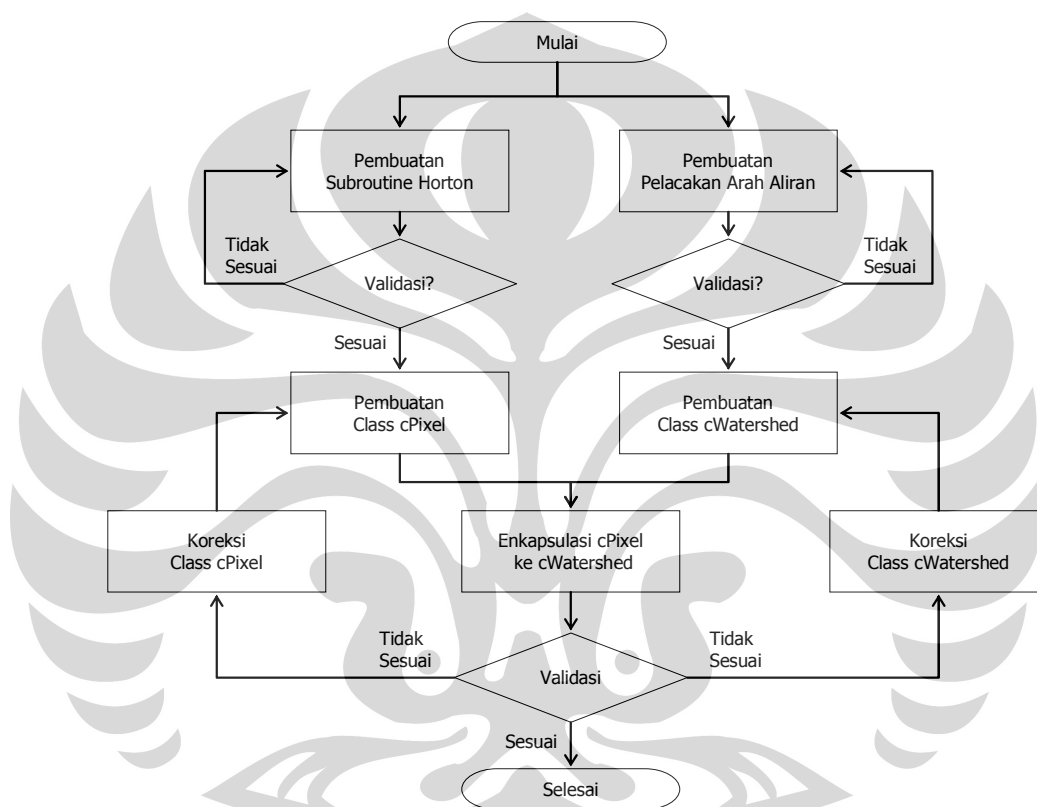
### 3.6.3. *PostGIS Processing*

*postGIS Processing* adalah proses dimana data-data yang telah siap dalam format *raw raster* kemudian dilakukan simulasi hujan-aliran sehingga didapatkan hasil akhir berupa besarnya limpasan permukaan dan besarnya infiltrasi yang terjadi pada wilayah studi. Perangkat tambahan yang akan dikembangkan dibuat berdasarkan *Horton Infiltration Excess Model* yang telah dibahas pada bab sebelumnya.

## 3.7. Metodologi Pelaksanaan Kajian

Pada sub-bab sebelumnya, telah dijelaskan mengenai kuantifikasi limpasan permukaan dengan menggunakan SIG yang dibagi menjadi tiga proses, yaitu *pre-GIS processing*, *GIS processing*, dan *post-GIS processing*. Dari ketiga rangkaian ini, tahapan *pre-GIS processing* dan *GIS processing* telah dilakukan pada kajian sebelumnya (Rizaldy, 2009). Sedangkan untuk *post-GIS processing* akan dibagi menjadi dua kajian besar, yaitu: pengembangan model dan aplikasinya.

Untuk pengembangan model dan validasinya akan dilakukan pada kajian ini, sedangkan aplikasi dan kalibrasi berdasarkan data-data wilayah studi yang telah disajikan pada kajian sebelumnya, akan dilakukan pada kajian berikutnya. Untuk pengembangan model dari sistem simulasi fisik hidrologi suatu kawasan itu sendiri, metodologi yang digunakan dalam pencapaian tujuan dalam kajian ini adalah seperti yang ditunjukkan pada gambar 3.12.



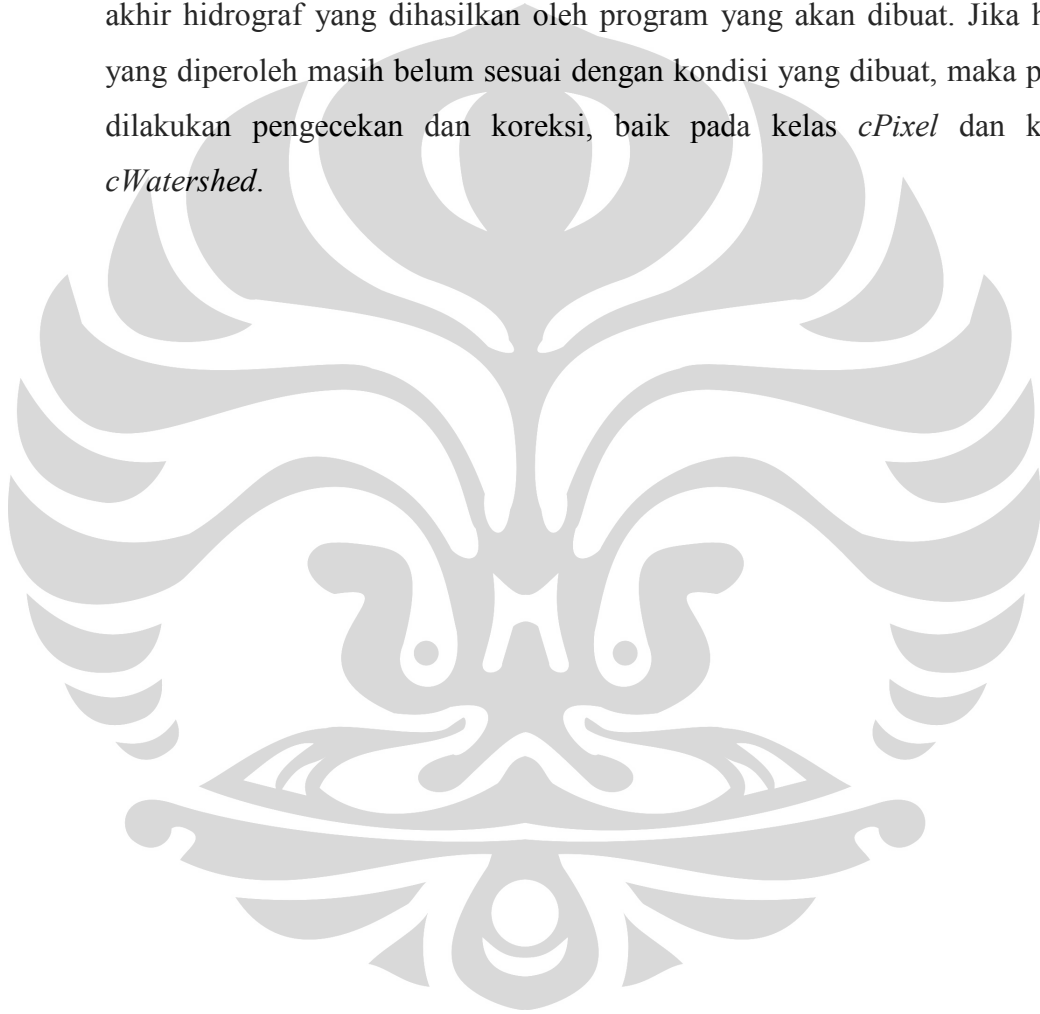
**Gambar 3.12.** Proses Utama Dalam Kajian

Dalam kajian ini akan dibuat dua buah kelas utama yaitu kelas *cPixel* dan *cWatershed*. Dalam proses simulasi hujan alir, proses yang paling penting adalah proses merubah hujan menjadi limpasan. Pada kajian ini digunakan model *horton infiltration excess*, dimana limpasan merupakan sisa dari infiltrasi. Oleh karena itu, sebelum subroutine horton dimasukkan kedalam kelas *cPixel*, harus divalidasi dulu hasil akhirnya. Begitu pula dengan pelacakan sistem aliran limpasan, harus dilakukan validasi sebelum dimasukkan kedalam kelas *cWatershed*, karena akan mempengaruhi bentuk akhir dari hidrograf yang dihasilkan.



Setelah kelas *cPixel* dan *cWatershed* siap dan telah dilakukan validasi terhadap subroutine penting. Barulah dilakukan enkapsulasi kelas *cPixel* kedalam kelas *cWatershed*, dimana nantinya kelas *cWatershed* akan mewadahi kelas-kelas *cPixel* yang saling terpisah, menjadi satu kesatuan yang masif dan saling terhubung satu sama lainnya.

Setelah proses enkapsulasi dilakukan dan kelas *cPixel* telah saling dihubungkan dengan pointer, maka perlu dilakukan validasi terhadap hasil akhir hidrograf yang dihasilkan oleh program yang akan dibuat. Jika hasil yang diperoleh masih belum sesuai dengan kondisi yang dibuat, maka perlu dilakukan pengecekan dan koreksi, baik pada kelas *cPixel* dan kelas *cWatershed*.



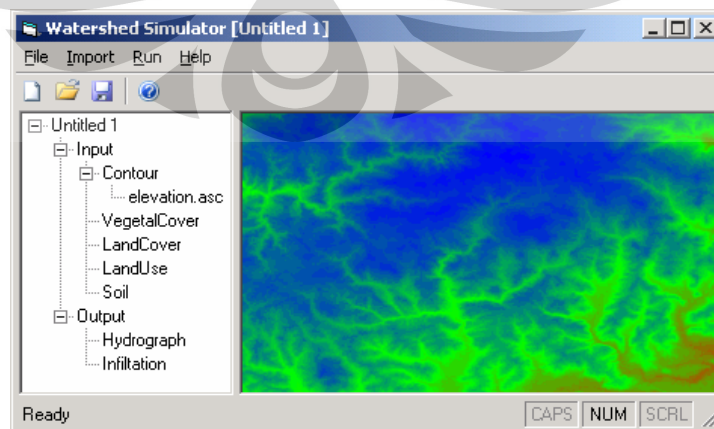
## BAB IV

### PENGEMBANGAN DAN VALIDASI MODEL

#### 4.1. Struktur Program

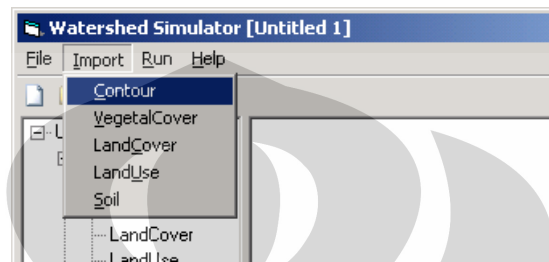
Model ini dikembangkan dengan menggunakan bahasa pemrograman Basic, dengan menggunakan IDE (*Integrated Development Environment*) Microsoft Visual Basic 6. Bahasa ini dipilih karena kemudahannya dalam merancang GUI (*Graphical User Interface*) secara visual sehingga pengembangan model ini akan dapat dilakukan dengan cepat, serta hasil akhirnya mudah digunakan. Penggunaan Basic sebagai bahasa pemrograman tingkat tinggi, membuat *source code* menjadi lebih mudah terbaca, sehingga saat terjadi kesalahan dan akan dilakukan *debugging* maupun ditujukan untuk pengembangan selanjutnya, dapat dilakukan dengan mudah.

Simulasi fisik hidrologi, dikerjakan oleh suatu modul khusus dan dapat di implementasikan ke berbagai program GIS sebagai *tools extension*. Namun untuk mengimplementasikan modul ini ke dalam program GIS, akan sangat spesifik dan tergantung kepada API yang disediakan oleh program tersebut. Oleh karena itu, dalam penulisan ini akan dibuat program terpisah yang bersifat *stand-alone*, untuk menjaga *source code* simulasi tetap orisinal sebelum di kompilasi ke berbagai program GIS.



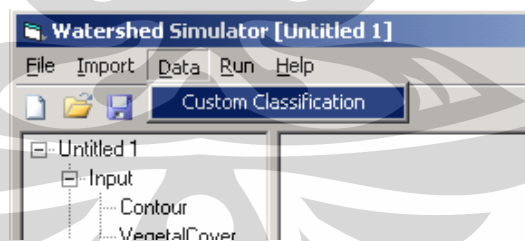
**Gambar 4.1** Tampilan Utama

Untuk melakukan simulasi fisik hidrologi menggunakan program ini, langkah awal yang harus dilakukan adalah memasukkan semua data peta raster berupa peta raster topografi, vegetal cover, land cover, land use, dan data tanah. Semua peta tersebut dimasukkan ke dalam program melalui perintah: *Import > Map Type*. Perhatikan gambar 4.2 untuk gambaran visual lebih jelas mengenai perintah ini.



**Gambar 4.2** Melakukan Import Raster Map

Setelah semua peta dimasukkan ke dalam program, jika ada data peta merupakan *custom classification*, maka perlu dilakukan klasifikasi kode data yang bersangkutan. Klasifikasi kode tanah tersebut dapat dilakukan dengan menggunakan perintah: *Data > Custom Classification*. Perhatikan gambar 4.3 untuk gambaran visual lebih jelas mengenai perintah ini. Penjelasan lebih lanjut mengenai *custom classification*, akan dijelaskan pada sub-bab berikutnya.



**Gambar 4.3** Pengklasifikasian Data Tanah.

Sebelum melakukan simulasi, perlu didefinisikan terlebih dahulu kejadian hujan (*rainstorm*) berupa tinggi hujan, durasi, dan tipe distribusi hujan, dengan menggunakan perintah: *Data > Rainstorm*. Barulah kemudian simulasi dapat dilakukan dengan mengeksekusi perintah *Run*. Setelah itu pointer akan berubah menjadi tanda positif (+) dan user diminta untuk melakukan klik pada titik *point of origin* dari DAS yang ingin dilakukan simulasi.

#### 4.1.1. Komponen Masukkan

Komponen masukkan atau *input* dari program ini secara umum dapat dikelompokkan menjadi dua kelompok besar, yaitu:

- data hujan, dan
- data parameter atau karakteristik DAS.

Data hujan yang dimasukkan berupa tinggi hujan, durasi hujan, dan tipe hujan, yang kemudian akan diproses oleh program untuk mensimulasikan proses hujan persatuan waktu. Sedangkan parameter atau karakteristik DAS yang dimasukkan mencakup berbagai data yang dibutuhkan untuk melakukan simulasi fisik hidrologi dalam suatu DAS, yaitu:

- topografi (elevasi),
- vegetal cover,
- land cover,
- land use,
- koefisien infiltrasi tanah ( $f_0$ ,  $f_c$ , dan  $k$ ).

Semua masukkan data parameter atau karakteristik DAS tersebut harus disajikan dalam bentuk file raster dengan format ascii, yang sudah dikonversi sebelumnya oleh program GIS. Struktur penyimpanan model data raster memiliki satu parameter untuk setiap pikselnya, sehingga untuk melakukan simulasi secara lengkap, dibutuhkan lima file raster dengan resolusi dan dimensi yang sama.

##### a. Data topografi.

Data masukan topografi, disajikan dalam bentuk data raster dimana setiap pikselnya mengandung nilai elevasi dari setiap pixel tersebut. Hal ini dibutuhkan untuk menentukan arah aliran (*flow path*) dari limpasan hujan yang mengalir dari piksel satu ke piksel lainnya dalam selang waktu.

80	74	62	45	45	34	39	56
80	74	74	62	45	34	39	56
74	74	62	62	45	34	39	39
62	62	45	45	34	34	34	39
45	45	45	34	34	30	34	39

**Gambar 4.4** Struktur Penyimpanan Data Elevasi

**b. Data *vegetal cover*.**

Data masukan *vegetal cover*, disajikan dalam bentuk data raster dimana setiap pixelnya mengandung nilai koefisien intersepsi dari setiap pixel tersebut. Hal ini dibutuhkan untuk mensimulasikan kejadian intersepsi hujan oleh vegetasi dalam suatu piksel. Nilai koefisien intersepsi pada dunia nyata akan sangat beragam dan tergantung kepada kepadatan dan jenis vegetasinya. Untuk mempermudah dalam pelaksanaan simulasi, berbagai jenis *vegeral cover* ini kemudian diklasifikasikan menjadi empat kode utama, yaitu:

- kode 0 untuk tanpa vegetasi,
- kode 1 untuk vegetasi dengan kerapatan rendah,
- kode 2 untuk vegetasi dengan kerapatan sedang, dan
- kode 3 untuk vegetasi dengan kerapatan tinggi.

Nantinya masing-masing kode tersebut akan mewakili suatu koefisien intersepsi berdasarkan tipenya, sebagai berikut:

- kode 0 mewakili nilai  $C_{li}$  sebesar 1,
- kode 1 mewakili nilai  $C_{li}$  sebesar 0.9,
- kode 2 mewakili nilai  $C_{li}$  sebesar 0.7, dan
- kode 3 mewakili nilai  $C_{li}$  sebesar 0.6.

Untuk implementasi pada berbagai program GIS, tidak menutup kemungkinan untuk memasukkan nilai koefisien intersepsi di luar kode tersebut, karena pada modul simulasi disediakan dua pilihan masukkan data, yaitu *code based* dan *custom coeficient*.

**c. Data *landcover*.**

Data masukan *landcover*, disajikan dalam bentuk data raster dimana setiap pixelnya mengandung nilai koefisien penutup lahan dari setiap pixel tersebut. Untuk mempermudah dalam pelaksanaan simulasi, berbagai jenis *vegeral cover* ini kemudian diklasifikasikan menjadi empat kode utama, yaitu:

- kode 0 untuk *Asphalt*,
- kode 1 untuk *Brick Paving*,

- kode 2 untuk *Roof Building*, dan
- kode 3 untuk *Grass Covered*.

Nantinya masing-masing kode tersebut akan mewakili suatu koefisien intersepsi berdasarkan tipenya, sebagai berikut:

- kode 0 mewakili nilai  $C_{lc}$  sebesar 0.80,
- kode 1 mewakili nilai  $C_{lc}$  sebesar 0.75,
- kode 2 mewakili nilai  $C_{lc}$  sebesar 0.85, dan
- kode 3 mewakili nilai  $C_{lc}$  sebesar 0.10.

Untuk implementasi pada berbagai program GIS, tidak menutup kemungkinan untuk memasukkan nilai koefisien penutup lahan di luar kode tersebut, karena pada modul simulasi disediakan dua pilihan masukkan data, yaitu *code based* dan *custom coeficient*.

#### **d. Data *landuse*.**

Data masukan *landuse*, disajikan dalam bentuk data raster dimana setiap pixelnya mengandung nilai koefisien penggunaan lahan dari setiap pixel tersebut. Untuk mempermudah dalam pelaksanaan simulasi, berbagai jenis *vegeral cover* ini kemudian diklasifikasikan menjadi delapan kode utama, yaitu:

- kode 0 untuk *Urban Bussiness*,
- kode 1 untuk *CommercialOffice*,
- kode 2 untuk *Residential tipe Homes*,
- kode 3 untuk *Residential tipe Condominium*,
- kode 4 untuk *Residential tipe Apartments*,
- kode 5 untuk *Suburban Residential*,
- kode 6 untuk *Light Industrial*,
- kode 7 untuk *Heavy Industial*,
- kode 8 untuk *Park* atau *Open Space*, dan
- kode 9 untuk *Rail* atau *Road Way*.

Nantinya masing-masing kode tersebut akan mewakili suatu koefisien intersepsi berdasarkan tipenya, sebagai berikut:

- kode 0 mewakili nilai  $C_{lu}$  sebesar 0.8,

- kode 1 mewakili nilai  $C_{lu}$  sebesar 0.6,
- kode 2 mewakili nilai  $C_{lu}$  sebesar 0.4,
- kode 3 mewakili nilai  $C_{lu}$  sebesar 0.5,
- kode 4 mewakili nilai  $C_{lu}$  sebesar 0.7,
- kode 5 mewakili nilai  $C_{lu}$  sebesar 0.3,
- kode 6 mewakili nilai  $C_{lu}$  sebesar 0.6,
- kode 7 mewakili nilai  $C_{lu}$  sebesar 0.8,
- kode 8 mewakili nilai  $C_{lu}$  sebesar 0.3,
- kode 9 mewakili nilai  $C_{lu}$  sebesar 0.2,

Untuk implementasi pada berbagai program GIS, tidak menutup kemungkinan untuk memasukkan nilai koefisien penggunaan lahan di luar kode tersebut, karena pada modul simulasi disediakan dua pilihan masukkan data, yaitu *code based* dan *custom coefficient*.

#### e. Data koefisien infiltrasi.

Data koefisien infiltrasi tanah ( $f_0$ ,  $f_c$ , dan  $k$ ), disajikan dalam bentuk data raster dimana setiap pixelnya mengandung nilai-nilai koefisien tersebut. Koefisien infiltrasi tanah ini terdiri dari tiga buah koefisien, hal ini mempersulit pengolahan karena struktur data raster hanya memuat satu nilai untuk tiap pikselnya. Untuk mengatasi ini, dilakukan pengkodean nilai-nilai koefisien berdasarkan jenis tanahnya secara default, yaitu:

**Tabel 4.1.** Klasifikasi Kode Jenis Tanah

Soil Type	Code
Sandy soils	
little or no vegetation	Kode 0
dense vegetation	Kode 1
poor developed	Kode 2
well developed	Kode 3
Loam soils	
little or no vegetation	Kode 4
dense vegetation	Kode 5
poor developed	Kode 6
well developed	Kode 7
Clay soils	
little or no vegetation	Kode 8
dense vegetation	Kode 9
poor developed	Kode 10
well developed	Kode 11

Nantinya masing-masing kode tersebut akan mewakili suatu koefisien intersepsi berdasarkan tipenya, sebagai berikut:

**Tabel 4.2.** Harga koefisien  $f_0$ ,  $f_c$ , dan  $k$

Kode	$f_0$ (mm/s)	$f_c$ (mm/s)	$k$ (s <sup>-1</sup> )
Kode 0	0.0353	0.0032	0.0055
Kode 1	0.0706	0.0036	0.0052
Kode 2	0.0172	0.0027	0.0050
Kode 3	0.0233	0.0030	0.0055
Kode 4	0.0212	0.0008	0.0107
Kode 5	0.0422	0.0011	0.0097
Kode 6	0.0117	0.0004	0.0083
Kode 7	0.0171	0.0007	0.0107
Kode 8	0.0071	0.0006	0.0022
Kode 9	0.0142	0.0008	0.0157
Kode 10	0.0056	0.0002	0.0017
Kode 11	0.0098	0.0005	0.0022

Untuk implementasi pada berbagai program GIS, tidak menutup kemungkinan untuk memasukkan nilai koefisien intersepsi di luar kode tersebut, karena pada modul simulasi disediakan dua pilihan masukkan data, yaitu *code based* dan *custom coeficient*.

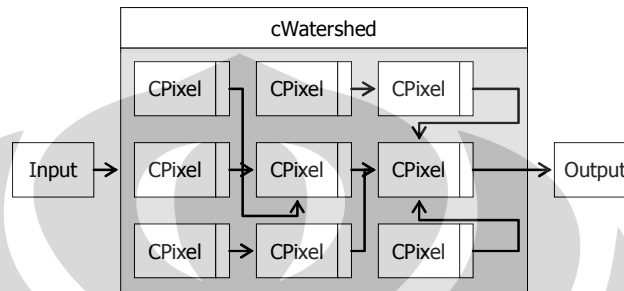
#### 4.1.2. Komponen Sistem

Program yang dibuat terdiri dari dua *class module*. *Class* pertama ditujukan untuk menangani sistem piksel tunggal dengan modul utamanya adalah *class cPixel*. Pada modul ini disimulasikan berbagai peristiwa fisik hidrologi berupa intersepsi, *landuse and landcover effect*, serta infiltrasi. Sehingga nantinya dari modul ini dihasilkan besarnya debit limpasan yang terjadi pada piksel tersebut dan nantinya akan dialirkan ke piksel lainnya yang memiliki elevasi lebih rendah daripada piksel tersebut.

Dalam sebuah sistem DAS yang lebih kompleks, tentu saja tidak hanya terdiri dari sebuah sistem piksel tunggal. Dalam sistem DAS tersebut terdapat ratusan bahkan ribuan sistem piksel yang saling terhubung satu sama lainnya. Oleh karena itu dibutuhkan sebuah *class* yang lebih besar, yang bertugas menyatukan ratusan sistem piksel tunggal tersebut menjadi sebuah sistem induk yang kompleks dan masif. Dalam kasus ini, sebuah sistem hujan-alir yang lebih kompleks ditangani oleh

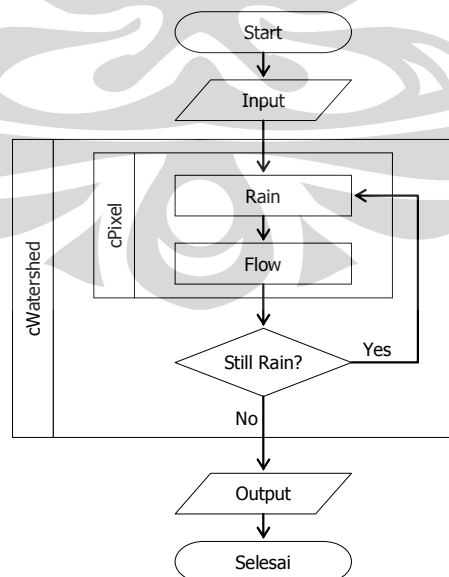


sebuah modul besar yang bernama *cWatershed*. Dimana kelas ini mewadahi semua sistem *cPixel* pada DAS, dan melakukan pelacakan terhadap arah aliran limpasan. Untuk gambaran lebih jelas mengenai sistem dan sub-sistem ini, perhatikan gambar 4.5 dimana kelas-kelas sistem tunggal *cPixel* saling terhubung satu sama lain dengan bantuan sistem induk *cWatershed*.



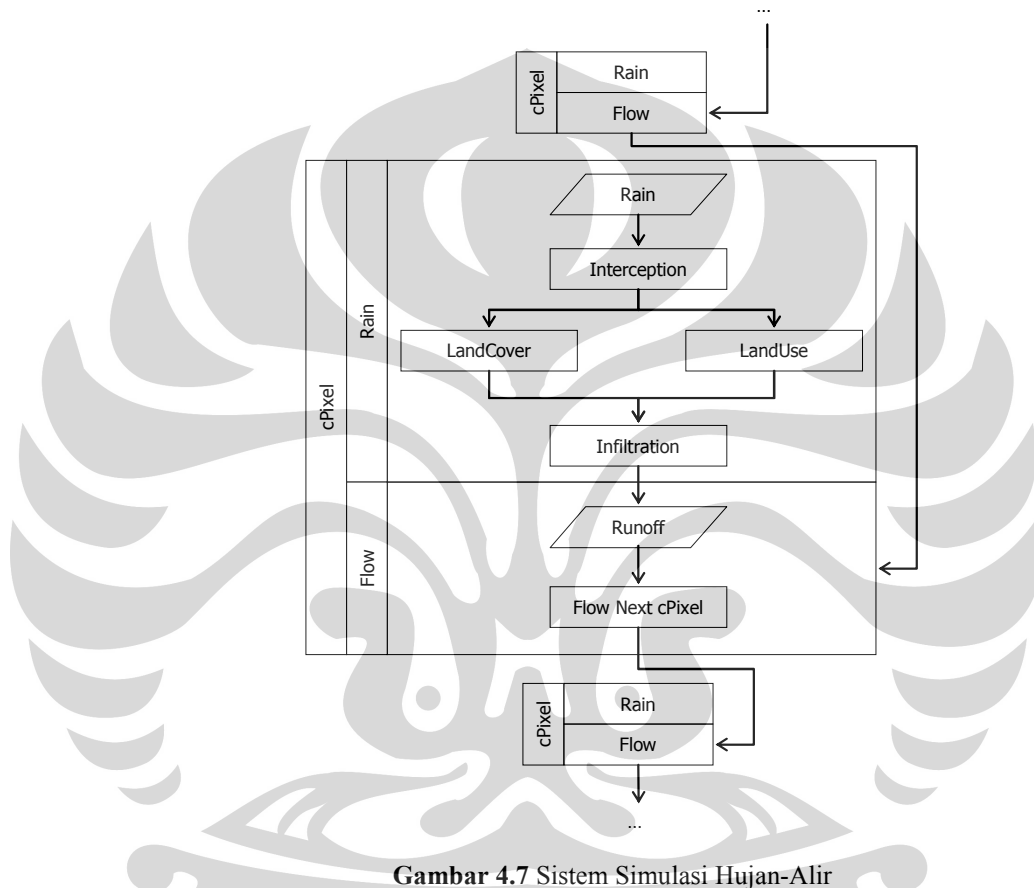
**Gambar 4.5** Rangkaian Sistem Simulasi

Lebih detail lagi, cara kerja kedua sistem ini diilustrasikan pada gambar 4.6, dimana pada sistem *cPixel* terdapat dua buah fungsi utama, yaitu fungsi hujan dan alir. Penggunaan keduanya diatur secara langsung oleh sistem *cWatershed*. Dengan kata lain, sistem *cWatershed* mengatur sistem *cPixel*, kapan harus hujan dan dialirkan sampai peristiwa hujan selesai. Fungsi hujan dan fungsi alir sendiri memiliki suatu alur sendiri yang kompleks yang akan dibahas selanjutnya.



**Gambar 4.6** Sistem Simulasi

Sistem hujan-alir yang saling terhubung dalam *class cPixel*, lebih detail prosesnya terlihat dalam gambar 4.7. Dari gambar tersebut dapat terlihat untuk fungsi hujan itu sendiri, terdiri dari beberapa peristiwa fisik meliputi: hujan, intersepsi, efek penutup dan penggunaan lahan, serta infiltrasi. Untuk detail perumusan dan algoritma berbagai peristiwa fisik ini telah dijelaskan pada bab sebelumnya.



Gambar 4.7 Sistem Simulasi Hujan-Alir

Yang menjadi sebuah perhatian besar dalam sistem simulasi hujan-alir ini adalah fungsi pelacakan aliran limpasan permukaan. Dalam kondisi ini terdapat beberapa kondisi yang memungkinkan terjadi kesalahan dalam melacak arah aliran yang menyebabkan suatu sistem DAS menjadi tidak sesuai dengan yang diharapkan.

Algoritma pelacakan aliran limpasan bekerja dengan melihat elevasi yang paling rendah pada piksel sekelilingnya dengan kedalaman pencarian satu level (piksel). Secara normal, pada algoritma ini akan ditemukan sebuah piksel dengan elevasi paling rendah, seperti yang

terlihat pada gambar 4.8, dimana limpasan permukaan akan mengalir akan mengalir ke piksel tersebut.

1	2	3
4	9	5
6	7	8

**Gambar 4.8** Pelacakan Aliran Limpasan

Namun saat dilakukan pelacakan, ada beberapa kondisi yang sangat sulit diprediksi arah alirannya. Hal ini dapat terjadi dikarenakan ditemukannya piksel terendah lebih dari satu dengan nilai elevasi yang sama, maupun saat kondisi piksel di sekeliling piksel yang ingin di lacak memiliki nilai elevasi yang sama dengan piksel utama (datar).

1	2	3
2	5	2
3	2	1

(a)

1	1	1
1	1	1
1	1	1

(b)

**Gambar 4.9** Kondisi Pelacakan yang Bermasalah

Dalam kondisi seperti ini, pelacakan aliran limpasan dilakukan dengan cara melakukan pencarian ke level berikutnya sampai maksimum 3 piksel di sekelilingnya. Jika dengan menggunakan cara tersebut masih tidak ditemukan solusi pengalirannya, maka akan ditentukan bahwa piksel tersebut mengalir ke piksel yang mengarah ke *point of origin* dari DAS.

2	3	4
3	1	5
4	5	6

**Gambar 4.10** Kondisi Pelacakan yang Bermasalah

Kondisi lain yang mengalami permasalahan adalah kondisi dimana piksel yang ingin dilacak arah alirannya, ternyata merupakan piksel

terendah dari piksel disekelilingnya. Dalam kondisi seperti ini, seharusnya pada kawasan tersebut membentuk genangan (*ponding*) karena membentuk cekungan. Namun karena kondisi seperti ini tidak dibahas karena batasan penulisan, maka jika ditemukan kondisi seperti ini akan ditentukan bahwa piksel tersebut tidak mengalir ke manapun, dengan kata lain piksel tersebut akan mengalir ke dirinya sendiri.

#### 4.1.3. Komponen Keluaran

Komponen keluaran utama dari model ini berupa grafik hidrograf aliran limpasan dari titik *point of origin* DAS. Yang kemudian dari grafik tersebut dapat ditentukan debit puncak dan waktu diperlukan untuk sampai ke debit puncak. Lebih lanjut lagi *user* dapat melakukan analisis dari hasil keluaran tersebut dengan melakukan berbagai skenario parameter dari DAS yang sedang ditinjau.

#### 4.2. Validasi Subroutine dan Output Model

Setelah dilakukan pembuatan model, maka perlu dilakukan validasi model yang telah dibuat. Validasi yang dilakukan berupa membandingkan hasil akhir program dengan beberapa kondisi atau skenario, sehingga didapatkan hasil yang logis sesuai dengan teori.

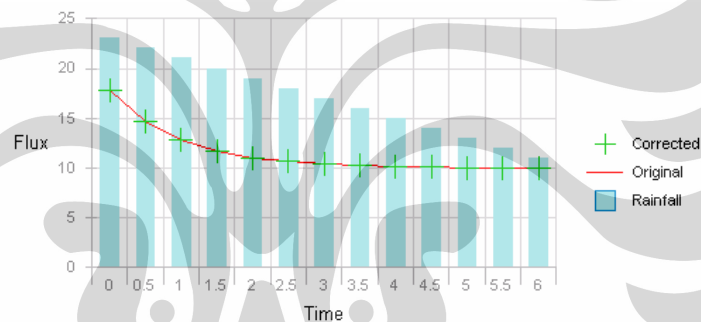
Validasi pada model ini akan dilakukan pada subroutine horton dan subroutine pelacakan arah aliran. Validasi pada kedua subroutine ini dirasa penting mengingat subroutine ini merupakan subroutine yang kritis mengingat semua sistem simulasi bergantung kepada kedua subroutine tersebut.

Hasil akhir atau output dari model juga perlu dilakukan validasi terhadap berbagai skenario karena hal ini akan menyangkut apakah hasil akhir hidrograf yang dihasilkan logis atau tidak. Sedangkan keakuratan nilainya sendiri nantinya akan dilakukan kalibrasi pada bahasan selanjutnya.

#### 4.2.1. Validasi Subroutine Horton

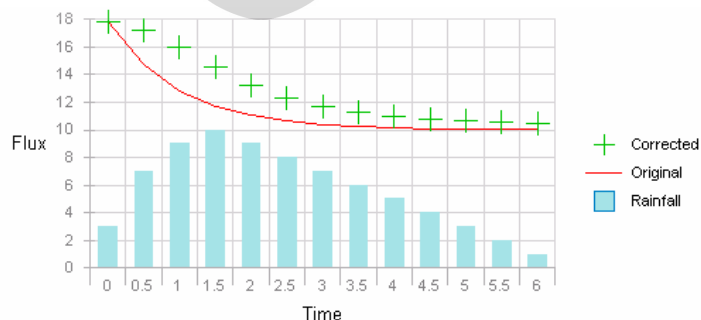
Validasi subroutine horton dilakukan dengan cara membandingkan grafik hasil plot fungsi laju infiltrasi horton secara matematis dengan hasil akhir yang didapat dengan menggunakan subroutine program yang dibuat. Jika grafik yang dihasilkan menunjukkan *trend* yang sama dengan hasil angka yang hampir serupa, maka program yang dibuat dinyatakan telah teruji kebenarannya.

Untuk skenario pertama, akan diuji subroutine horton berdasarkan kondisi intensitas hujan yang terjadi selama dt selalu lebih besar daripada laju infiltrasi tanah. Dapat terlihat dalam gambar 4.11 dalam kondisi ini trend dan hasil yang didapat tepat mengikuti fungsi laju infiltrasi horton. Ini menunjukkan bahwa besarnya laju infiltrasi tidak terjadi perubahan dan tetap mengikuti fungsi horton di awal.



Gambar 4.11 Hasil Pengujian Skenario 1

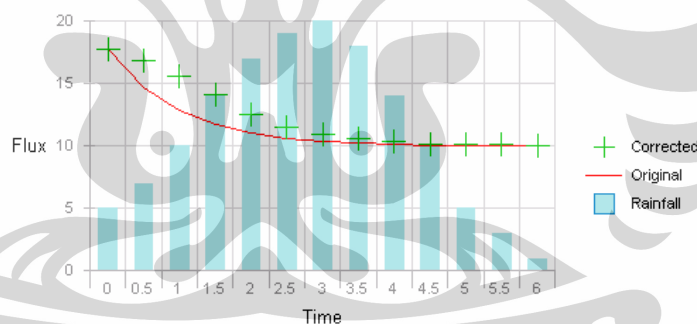
Untuk skenario kedua, akan diuji subroutine horton berdasarkan kondisi intensitas hujan yang terjadi selama dt selalu lebih kecil daripada laju infiltrasi tanah. Dapat terlihat dalam gambar 4.12 dalam kondisi ini trend yang dihasilkan masih sama, namun nilai yang dihasilkan berbeda.



Gambar 4.12 Hasil Pengujian Skenario 2

Hal ini sangat logis jika hasil akhir dari program tidak mengikuti kaidah fungsi horton. Saat intensitas hujan kurang dari laju infiltrasi, maka tentu saja secara logis besarnya yang terinfiltrasi selama dt bukan sebesar laju infiltrasinya, melainkan sebesar intensitas hujan tersebut. Hal ini secara otomatis akan mengakibatkan kumulatif volume yang sudah teresap akan menjadi lebih kecil dari yang seharusnya sehingga terjadi peningkatan laju infiltrasi, sehingga kurvanya akan berada diatas kurva horton yang sebenarnya.

Untuk skenario ketiga, akan diuji subroutine horton berdasarkan kondisi intensitas hujan yang terjadi selama dt bervariasi terhadap laju infiltrasi tanah. Dapat terlihat dalam gambar 4.13 dalam kondisi ini trend yang dihasilkan masih sama, namun terlihat pada kondisi awal, terjadi kenaikan nilai laju infiltrasi yang disebabkan oleh fenomena yang telah dijelaskan sebelumnya. Namun saat Intensitas hujan mulai naik diatas kurva laju infiltrasi, dapat terlihat nilai horton perlahan mulai turun dan hampir mengikuti lagi fungsi horton awal.

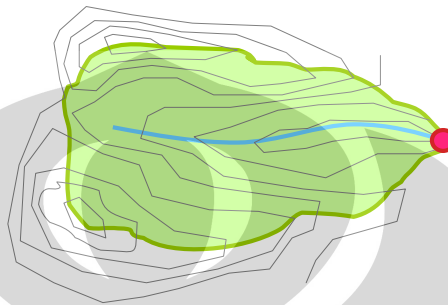


**Gambar 4.13** Hasil Pengujian Skenario 3

Pengujian subroutine horton pada tiga kondisi tersebut, telah menunjukkan hasil yang logis menurut fenomena fisik yang terjadi didalamnya. Hal ini cukup untuk membuktikan bahwa algoritma tersebut telah siap dan dapat dimasukkan kedalam sistem simulasi hidrologi yang lebih besar.

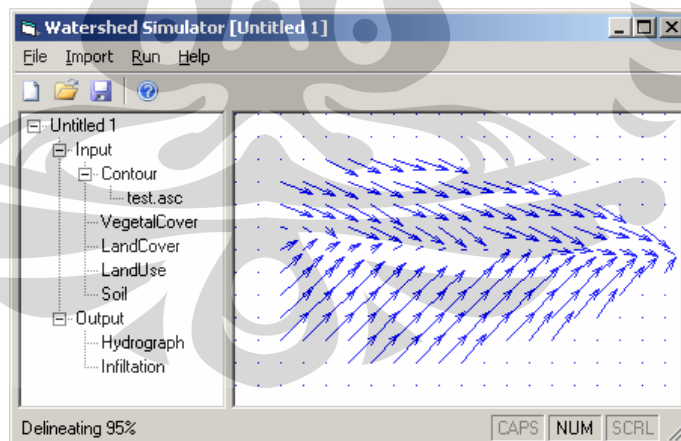
#### 4.2.2. Validasi Subroutine Pelacakan Arah Aliran

Validasi subroutine pelacakan arah aliran akan dilakukan satu kali, yaitu dengan membuat sebuah peta topografi hipotetik dengan bentuk kontur yang jelas dan dapat dibayangkan arah alirannya, hal ini dimaksudkan untuk memudahkan pengecekan apakah arah aliran yang dihasilkan sesuai dengan keadaan yang logis atau tidak.



**Gambar 4.14** Kontur Pengujian Penacakan Aliran

Kontur hipotetik yang digunakan adalah kotur seperti yang terlihat pada gambar 4.14. Dalam kondisi seperti ini, secara logis seharusnya arah aliran akan menuju ke tengah (sungai) kemudian mengalir menuju *point of origin* (titik outlet). Setelah dilakukan running, maka didapat hasil akhir dari pelacakan arah aliran ini seperti yang ditunjukkan pada gambar 4.15.



**Gambar 4.15** Hasil Pelacakan Arah Aliran

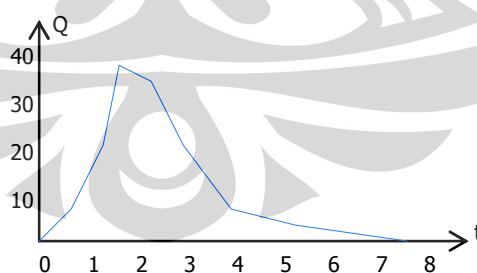
Dari pengujian algoritma pelacakan arah aliran yang telah dilakukan, dapat terlihat arah alirannya telah menunjukkan hasil yang logis dan dapat diandalkan. Seperti yang terlihat pada gambar 4.15 dimana garis-garis berpanah menunjukkan arah aliran limpasan sampai menuju

outlet sesuai dengan ekpektasi yang logis. Hal ini cukup untuk membuktikan bahwa subroutine tersebut telah siap dan dapat dimasukkan kedalam sistem simulasi hidrologi yang lebih besar.

#### 4.2.3. Validasi Output Hidrograf

Validasi terakhir dan yang paling penting adalah menguji hasil akhir dari program berupa Hidrograf pada *point of origin* (outlet). validasinya pada hasil akhir hidrograf ini mutlak diperlukan, karena nantinya jika program ini dipakai oleh berbagai user sebagai suatu alat bantu (*tools*) dalam melakukan analisa, hasil akhir hidrograf tersebut dapat dipertanggungjawabkan kebenarannya. Validasi akan dilakukan dengan cara melakukan pengujian terhadap perubahan parameter dan melihat bentuk dan serta perubahannya akibat perubahan parameter tersebut.

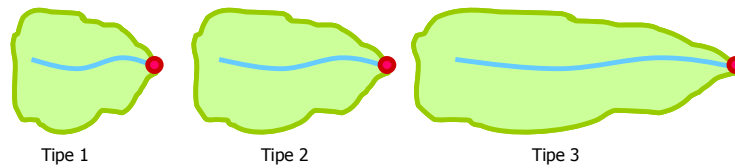
Pengujian pertama dilakukan untuk melihat bentuk hasil akhir hidrograf. Pada pengujian ini dibuat sebuah DAS sederhana, dengan *vegetal cover*, tataguna lahan, penutup lahan, dan jenis tanah yang seragam. Setelah itu dilakukan simulasi dengan intensitas hujan konstan selama 1 menit. Setelah dilakukan proses simulasi didapatlah hidrograf seperti pada gambar 4.16 pada *point of origin*. Bentuk hidrograf outlet pada saat pengujian ini telah sesuai dengan bentuk hidrograf umumnya, dan siap untuk dilakukan pengujian terhadap perubahan parameter.



Gambar 4.16 Hasil Akhir Hidrograf

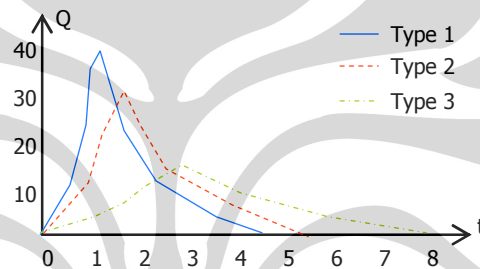
Pengujian kedua dilakukan untuk menguji bentuk akhir hidrograf akibat perubahan parameter bentuk DAS. Pada pengujian ini dibuat tiga tipe DAS dengan bentuk yang berbeda, namun dengan *vegetal cover*, tataguna lahan, penutup lahan, dan jenis tanah yang seragam.





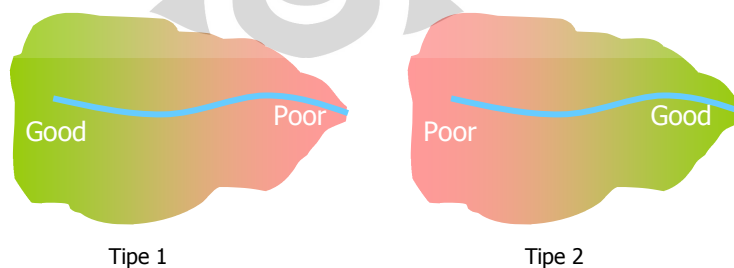
**Gambar 4.17** Perubahan Parameter Bentuk DAS

Kemudian dilakukan simulasi dengan intensitas hujan konstan selama 1 menit. Setelah dilakukan proses simulasi, didapatkan hidrograf seperti pada gambar 4.18 pada *point of origin* untuk ketiga tipe DAS. Bentuk hidrograf yang dihasilkan telah sesuai dan logis dimana untuk tipe pertama dengan bentuk DAS lebih pendek, menghasilkan hidrograf dengan *time to peak* yang lebih singkat dibandingkan dengan bentuk DAS yang lebih panjang (tipe dua dan tiga).



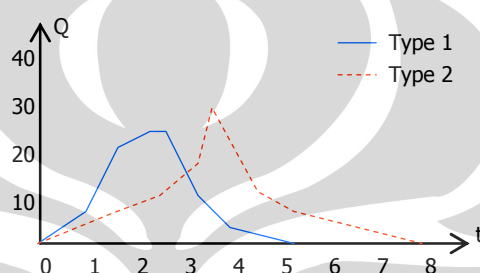
**Gambar 4.18** Perbandingan Hidrograf Akibat Bentuk DAS

Pengujian selanjutnya dilakukan untuk menguji bentuk akhir hidrograf terhadap perubahan parameter nilai koefisien penutup lahan dan penggunaan lahan. Pada pengujian ini dibuat dua DAS dengan bentuk yang tetap, *vegetal cover*, dan jenis tanah yang seragam, namun terdapat gradasi perubahan penutup lahan dan penggunaan lahan seperti yang terlihat pada gambar 4.19.



**Gambar 4.19** Perubahan Parameter Penutup Lahan dan Tata Guna Lahan

Kemudian dilakukan simulasi dengan intensitas hujan konstan selama 3 menit. Setelah dilakukan proses simulasi, didapatlah hidrograf seperti pada gambar 4.20 pada *point of origin* untuk kedua tipe DAS. Untuk tipe pertama, dimana peralihan penggunaan dan penutup lahan perlahan berubah dari baik ke buruk, dapat terlihat hasil hidrograf akhirnya berbentuk cukup logis, dimana diawal kenaikan debit cukup signifikan karena limpasan yang besar di hulu DAS, sedangkan saat mencapai debit puncak perlahan melandai karena kontribusi debit limpasan di hulu sangat sedikit akibat penataan lahan yang baik.



**Gambar 4.20** Perbandingan Hidrograf Akibat Bentuk DAS

Untuk kondisi kedua, dimana peralihan penggunaan dan penutup lahan perlahan berubah dari buruk ke baik, juga terlihat hasil hidrograf akhirnya berbentuk cukup logis, dimana diawal kenaikan debit limpasan cukup landai karena diawal kontribusi debit limpasan di hulu cukup sedikit, sedangkan saat mencapai debit puncak, kenaikan debit limpasan bertambah cukup signifikan karena debit limpasan di hulu yang cukup mulai berkontribusi ke outlet ke outlet.

#### 4.3. Validitas Model

Setelah dilakukan berbagai pengujian dan validasi terhadap model yang sedang dikembangkan. Program simulasi menunjukkan hasil validitas yang cukup baik, sehingga program ini dinyatakan layak dan siap untuk digunakan sebagai alat bantu simulasi. Nantinya program ini juga dapat dilakukan aplikasi langsung di wilayah studi, tentunya setelah dilakukan kalibrasi hasil akhir yang di peroleh pada program ini.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1. Kesimpulan

Program yang sedang dikembangkan dalam kajian ini dapat melakukan simulasi fisik hidrologi pada suatu kawasan (DAS) dimana data-data input yang dibutuhkan baik data primer maupun data sekunder, merupakan hasil konversi *software* GIS dari berbagai format file yang berbeda menjadi format yang diseragamkan dan dipadatkan menjadi piksel-piksel dalam skala kecil dan dalam jumlah yang masif, dengan resolusi yang diizinkan sebesar 0.25-1 m untuk setiap piksel, dengan permukaan yang tidak terlalu datar dan banyak cekungan..

Simulasi hujan-alir yang dilakukan oleh program ini dijalankan berdasarkan model klasik *Horton Infiltration Excess Model* dimana limpasan merupakan sisa dari infiltrasi dengan memasukkan berbagai fenomena lain berupa intersepsi, efek dari penutup dan penggunaan lahan, juga efek pergerakan aliran berdasarkan ketinggian, sehingga model yang dihasilkan menjadi lebih dinamis dari model klasik sebelumnya.

Proses infiltrasi yang menjadi cakupan utama dalam model ini dikembangkan berdasarkan fungsi kurva infiltrasi horton yang kemudian disesuaikan dan dikembangkan pada berbagai kondisi yang kemudian dirubah menjadi program komputer. Penyesuaian ini berupa koreksi laju infiltrasi akibat intensitas yang lebih kecil dari laju infiltrasi sehingga menyebabkan infiltrasi kumulatif menjadi berkurang dan laju infiltrasi menjadi bertambah. Penyesuaian ini kemudian diuji dan hasil akhirnya menunjukkan bahwa respon yang diperoleh sudah sesuai dengan apa yang diharapkan.

Proses aliran limpasan yang dinamis didapat dengan mencari pergerakan limpasan berdasarkan gravitasi, dimana limpasan mengalir dari elevasi tinggi menuju ke elevasi yang lebih rendah. Untuk kondisi yang logis, respon yang dihasilkan sesuai dengan apa yang diharapkan. Namun

pada tahapan ini, banyak sekali kondisi yang menimbulkan kekeliruan dalam pelacakan arah aliran, sehingga perlu dilakukan beberapa asumsi pada kondisi tertentu agar validitas hasil akhir tetap terjaga.

Pengujian model secara hipotetikal dilakukan untuk melihat respon yang dihasilkan berdasarkan perubahan parameter bentuk DAS dan tata guna lahan pada DAS hipotetik. Setelah melalui berbagai pengujian, program menunjukkan hasil yang sesuai dan logis berdasarkan fenomena alam yang ada, sehingga program ini dinyatakan siap untuk langsung dilakukan uji coba terhadap kondisi aslinya dilapangan berupa penerapan pada wilayah studi pada kajian selanjutnya.

## 5.2. Saran

Diperlukan pengembangan lebih lanjut mengenai efek dari tata guna lahan (*land use*) dan penutup lahan (*land cover*) terhadap limpasan yang dihasilkan oleh program ini. Interaksi antara kedua komponen tersebut dalam simulasi proses fisik pada kajian ini, hanya merupakan penyederhanaan dari proses yang lebih kompleks, sehingga output yang dihasilkan masih belum sesuai dengan kondisi aslinya di alam.

Pada algoritma pelacakan arah aliran, masih terdapat berbagai kesalahan pada kondisi-kondisi tertentu yang mengakibatkan arah aliran menjadi keliru. Kesalahan dalam pelacakan arah aliran ini tentunya dapat menyebabkan kesalahan dalam pembentukan sistem DAS yang mengakibatkan bentuk hidrograf yang dihasilkan melenceng jauh dari yang di harapkan dan hasil akhir yang dihasilkan menjadi kurang presisi.

Diperlukannya penerapan langsung ke wilayah studi pada program ini, sehingga nantinya dapat terlihat dengan jelas hasil akhir yang diperoleh pada program ini (hidrograf di *point of origin*), dan dapat dilakukan kalibrasi terhadap hasil akhir tersebut pada kondisi sebenarnya di wilayah tersebut. Dengan dilakukannya penerapan pada wilayah studi ini, akan terlihat arah pengembangan lebih lanjut dari program ini nantinya sehingga kedepan tingkat akurasi dan presisi dari program ini menjadi semakin baik.

## DAFTAR PUSTAKA

- Rizaldy, “*Proses Digital Sebagai Alat Bantu Sistem Informasi Pengelolaan Air Hujan Berbasis Low Impact Development*”, Depok: Skripsi Sarjana, Teknik Sipil, Program Sarjana Teknik Sipil, Fakultas Teknik, Universitas Indonesia Depok, 2009.
- C.D. Soemarto, *Hidrologi Teknik* (Jakarta : Erlangga, 1999) DR. Ir. Suyono Sosrodarsono, *Hidrologi Untuk Pengairan* (Jakarta : PT. Pradnya Paramita, 1983)
- DR. Ir. Suyono Sosrodarsono, and Dr. Maseteru Tominaga, *Perbaikan dan Pengaturan Sungai*, terj. Ir. M. Yusuf Gayo, dkk. (Jakarta : PT. Pradnya Paramita, 1984)
- Pahlevi, “*Perencanaan Sistem Drainase Depok Berdasarkan Hidrologi*”, Depok: Skripsi Sarjana, Teknik Sipil, Program Sarjana Teknik Sipil, Fakultas Teknik, Universitas Indonesia Depok, 2006.
- Ersin Seyhan, *Dasar-Dasar Hidrologi* (Yogyakarta : Gadjah Mada University Press, 1990)
- Ferdinanto, “*Kajian Pengaruh Tata Guna Lahan Terhadap Sistem Drainase Makro Pada Wilayah Perkotaan*”, Depok: Skripsi Sarjana, Teknik Sipil, Program Sarjana Teknik Sipil, Fakultas Teknik, Universitas Indonesia Depok, 2007.
- Imam Subarkah, *Hidrologi Untuk Perencanaan Bangunan Air* (Bandung : Idea Dharma, 1980)
- Linsley, Ray K, *Hidrologi Untuk Insinyur* (Jakarta : Erlangga, 1995) Ven Te Chow, David R. Maidment, Larry W. Mays, *Applied Hydrology*, (Singapore, McGraw-Hill Book, 1988)
- Chay Asdak, *Hidrologi dan Pengelolaan Daerah Aliran Sungai* (Gadjah Mada University Press, 2002)
- David R. Maidment, *Handbook of Hydrology* (McGraw-Hill, Inc. 1992)
- A. Longley, Paul, Michael F. Goodchild, David J. Maguire, and David W. Rhind. *Geographic Information Systems and Science*. West Sussex, England: John Wiley & Sons Ltd, 2001.
- Chang, Kang -Tsung. *Introdcution To Geographic Information Systems*. New York: McGraw-Hill, 2002.
- Economic and Social Comminssion for Asia and the Pasific. *Manual on GIS for Planner and Decision Makers*. New York: United Nations, 1996.
- Environmental Systems Research Institute (ESRI), Inc. *ESRI.Com*. 2006. [www.esri.com](http://www.esri.com) (accessed March 12, 2007).
- Gumelar, Dhani. *Implemantasi Kelompok Data Dasar dalam Penentuan Kawasan Lindung (Studi Kasus Pembangunan IDSD Provinsi Jawa Barat)*. Bandung: Tesis Magister, Bidang Geomatika, Program Magister Teknik

Geodesi, Fakultas Teknik Sipil dan Perencanaan, Institut Teknologi Bandung, 2004.

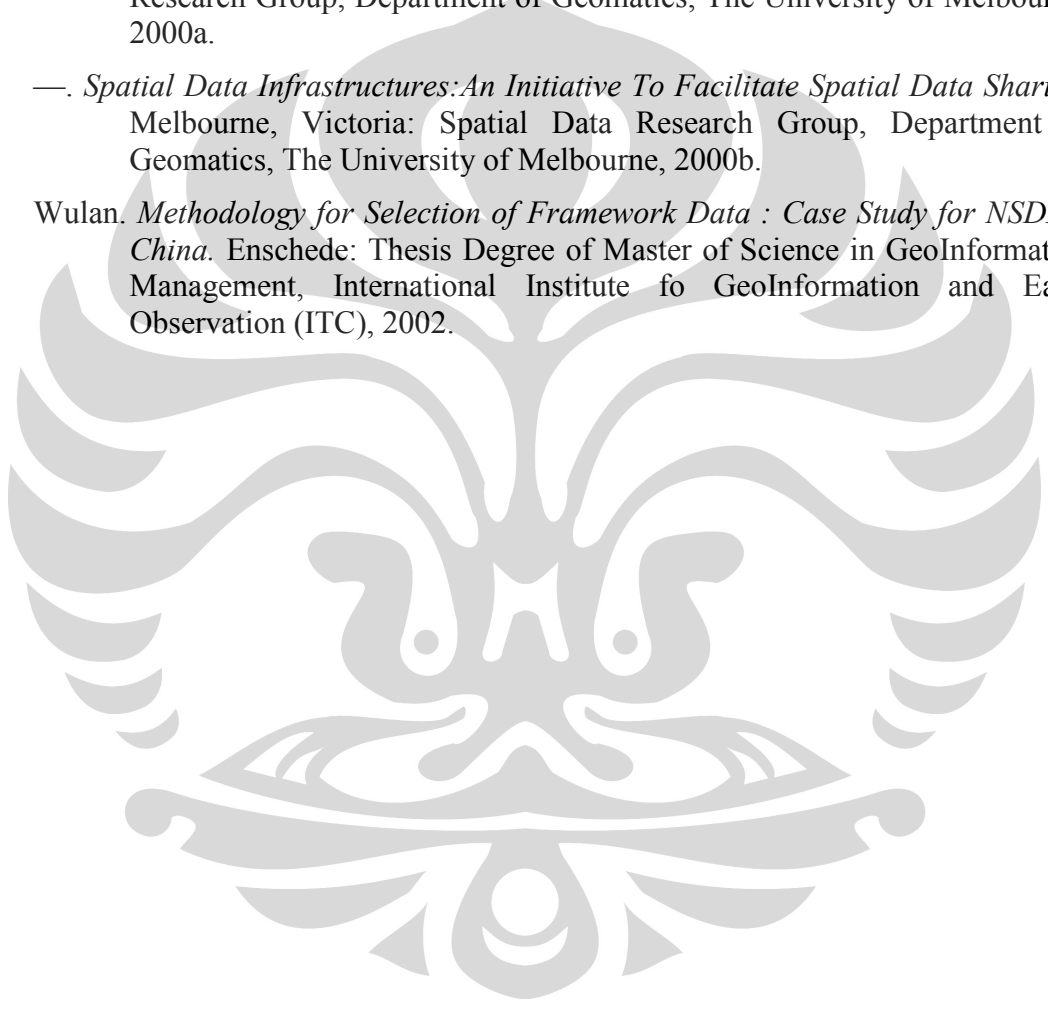
Prahasta, Eddy. *Konsep-konsep Dasar Sistem Informasi Geografis*. Bandung: Informatika, 2001.

Radjabidfard, Abbas. *SDI Hierarchy, from Local to Global SDI Initiatives*. Melbourne, Victoria: Spatial Data Research Group, Departement of Geomatics. The University of Melbourne, 2001.

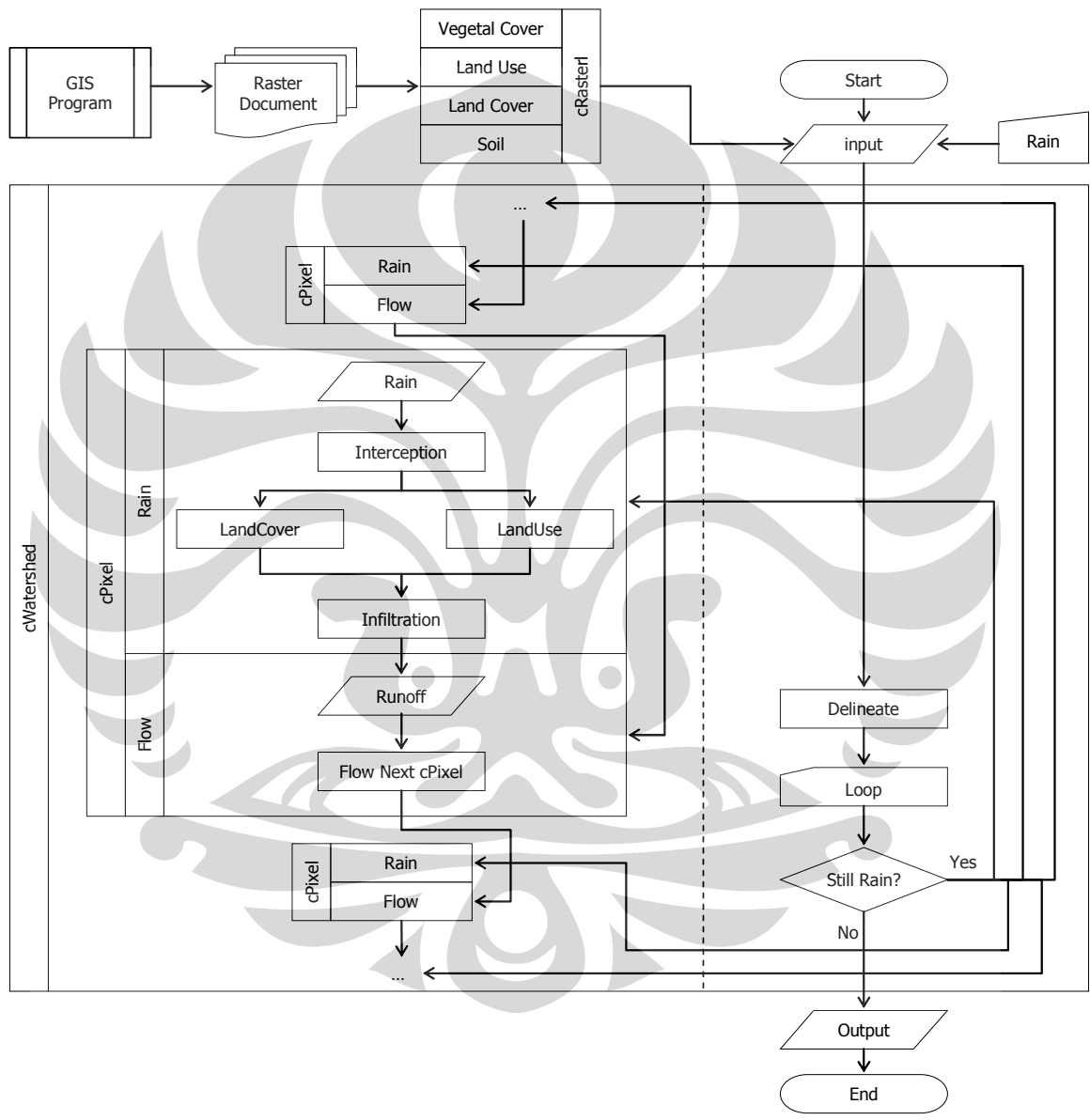
Rajabidfard, Abbas, and I.P. Williamson. "Spatial Data Infrastructures : Concept, SDI Hierarchy and Future Directions." Melbourne, Victoria: Spatial Data Research Group, Department of Geomatics, The University of Melbourne, 2000a.

—. *Spatial Data Infrastructures: An Initiative To Facilitate Spatial Data Sharing*. Melbourne, Victoria: Spatial Data Research Group, Department of Geomatics, The University of Melbourne, 2000b.

Wulan. *Methodology for Selection of Framework Data : Case Study for NSDI in China*. Enschede: Thesis Degree of Master of Science in GeoInformation Management, International Institute fo GeoInformation and Earth Observation (ITC), 2002.



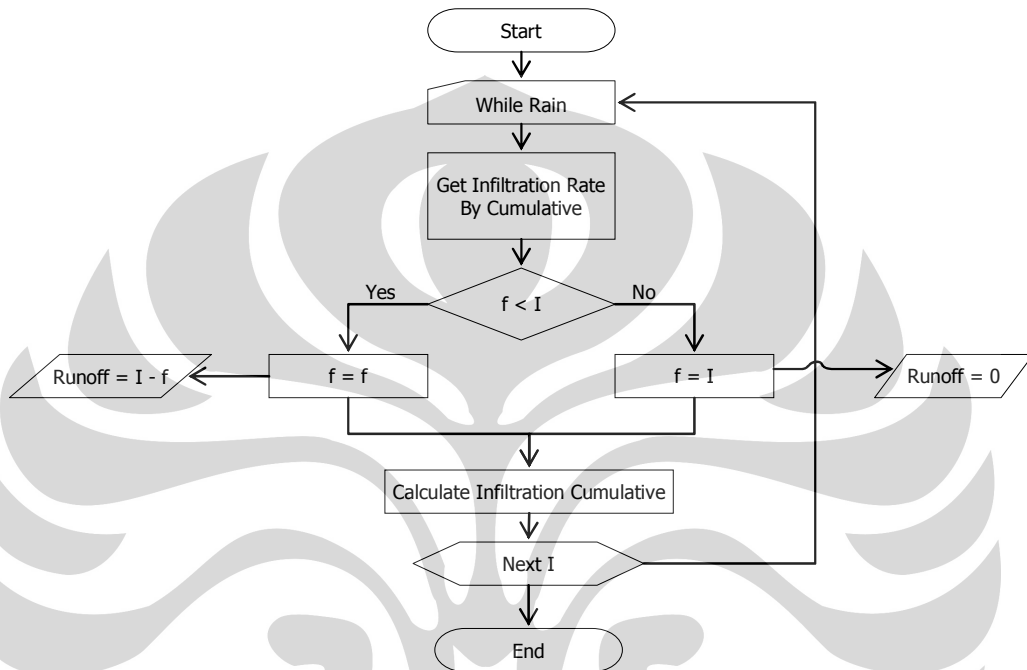
# LAMPIRAN 1 STRUKTUR PROGRAM



## LAMPIRAN 2

### HORTON INFILTRATION EXCESS

#### 1. Algoritma Horton Infiltration Excess



#### 2. Subroutine Horton Dalam cPixel

```

0001 Public Enum eSoilType
0002     Sandy_LitteVegetation = 0
0003     Sandy_DenseVegetation = 1
0004     Sandy_PoorDeveloped = 2
0005     Sandy_WellDeveloped = 3
0006     Loam_LitteVegetation = 4
0007     Loam_DenseVegetation = 5
0008     Loam_PoorDeveloped = 6
0009     Loam_WellDeveloped = 7
0010     Clay_LitteVegetation = 8
0011     Clay_DenseVegetation = 9
0012     Clay_PoorDeveloped = 10
0013     Clay_WellDeveloped = 11
0014 End Enum
0015
0016 Private iCumul      As Double 'in mm
0017 Private iTime      As Double 'in sec
0018 Private f0         As Double 'in mm/sec
0019 Private fc         As Double 'in mm/sec
0020 Private k          As Double 'in 1/sec
  
```



```

0021 Private Q As Double 'in m^3/s
0022
0023 Public Sub HortonExcess(ByVal I As Double, ByVal dt As Double)
0024     Dim f1 As Double
0025     Dim f2 As Double
0026
0027     '// Infiltration
0028     f1 = f(iTime)
0029     f2 = f(iTime + dt)
0030
0031     If f1 < I Then
0032         ' No Correction Needed
0033         iCumul = iCumul + (f1 + f2) / 2 * dt
0034
0035         '// Runoff (Infiltration Excess)
0036         Q = (I - (f1 + f2) / 2) / 1000 * A
0037     Else
0038         ' Correction Needed
0039         If f2 < I Then
0040             ' Intersecting
0041             Dim mdt As Double
0042             Do While (I - f(iTime + mdt)) > 0.00001
0043                 mdt = mdt + (I - f(iTime + mdt)) / (-k * (f0 - f -
0044 fc) * Exp(iTime + mdt))
0045             Loop
0046
0047             ' Calculating Start to Mid
0048             iCumul = iCumul + I * mdt
0049             iTime = t(iCumul)
0050
0051             ' Calculating Mid To End
0052             f1 = f(iTime)
0053             f2 = f(iTime + (dt - mdt))
0054             iCumul = iCumul + (f1 + f2) / 2 * (dt - mdt)
0055
0056             '// Runoff (Infiltration Excess)
0057             Q = (I - (f1 + f2) / 2) / 1000 * A
0058         Else
0059             ' No Intersecting
0060             iCumul = iCumul + I * dt
0061
0062             '// Runoff (Infiltration Excess)
0063             Q = 0
0064         End If
0065     End If
0066
0067     ' Adjust Time
0068     iTime = t(iCumul)
0069 End Sub
0070
0071
0072 '////////////////////////////////////
0073 '// Horton Infiltration Procedure and Funcions
0074
0075 ' Get Infiltration Rate by Time

```

```

0076 Private Function f(ByVal t As Double) As Double
0077     f = fc + (f0 - fc) * Exp(-k * t)
0078 End Function
0079
0080 ' Get Infiltration Cumulative by Time
0081 Private Function Cumul(ByVal t As Double) As Double
0082     Cumul = fc * t + (f0 - fc) / k * (1 - Exp(-k * t))
0083 End Function
0084
0085 ' Get Infiltration Time by Cumulative
0086 Private Function t(ByVal Cumulative As Double) As Double
0087     Do While Abs(Cumulative - Cumul(t)) > 0.00001
0088         t = t + (Cumulative - Cumul(t)) / f(t)
0089     Loop
0090 End Function
0091
0092
0093 '////////////////////////////////////
0094 '// Class Properties
0095 Friend Property Let Runoff(ByVal Value As Double)
0096     Q = Value
0097 End Property
0098
0099 Public Property Get Runoff() As Double
0100     Runoff = Q
0101 End Property
0102
0103 Public Property Let SoilType(ByVal Value As eSoilType)
0104     Select Case Value
0105         Case Sandy_LitteVegetation
0106             f0 = 0.0353: fc = 0.0032: k = 0.0055
0107         Case Sandy_DenseVegetation
0108             f0 = 0.0706: fc = 0.0036: k = 0.0052
0109         Case Sandy_PoorDeveloped
0110             f0 = 0.0172: fc = 0.0027: k = 0.0051
0111         Case Sandy_WellDeveloped
0112             f0 = 0.0233: fc = 0.0031: k = 0.0055
0113         Case Loam_LitteVegetation
0114             f0 = 0.0212: fc = 0.0008: k = 0.0107
0115         Case Loam_DenseVegetation
0116             f0 = 0.0422: fc = 0.0011: k = 0.0097
0117         Case Loam_PoorDeveloped
0118             f0 = 0.0117: fc = 0.0004: k = 0.0083
0119         Case Loam_WellDeveloped
0120             f0 = 0.0171: fc = 0.0007: k = 0.0107
0121         Case Clay_LitteVegetation
0122             f0 = 0.0071: fc = 0.0006: k = 0.0022
0123         Case Clay_DenseVegetation
0124             f0 = 0.0141: fc = 0.0008: k = 0.0157
0125         Case Clay_PoorDeveloped
0126             f0 = 0.0056: fc = 0.0002: k = 0.0017
0127         Case Clay_WellDeveloped
0128             f0 = 0.0098: fc = 0.0005: k = 0.0022
0129     End Select
0130 End Property

```

```

0131
0132 Public Sub SoilProperties(ByVal cf0 As Double, ByVal cfc As Doub _
0133 ble, ByVal ck As Double)
0134     f0 = cf0
0135     fc = cfc
0136     k = ck
0137 End Sub
0138
0139 Public Property Let Soil_f0(ByVal Value As Double)
0140     f0 = Value
0141 End Property
0142
0143 Public Property Get Soil_f0() As Double
0144     Soil_f0 = f0
0145 End Property
0146
0147 Public Property Let Soil_fc(ByVal Value As Double)
0148     fc = Value
0149 End Property
0150
0151 Public Property Get Soil_fc() As Double
0152     Soil_fc = fc
0153 End Property
0154
0155 Public Property Let Soil_k(ByVal Value As Double)
0156     k = Value
0157 End Property
0158
0159 Public Property Get Soil_k() As Double
0160     Soil_k = k
0161 End Property

```

### 3. Hasil Validasi Subroutine

Validasi dilakukan dengan nilai parameter tanah sebesar:

- f0 sebesar 20 mm/sec,
- fc sebesar 10 mm/sec, dan
- k sebesar 1 sec<sup>-1</sup>;

Dengan intensitas hujan bervariasi sesuai dengan jenis pengujian yang dilakukan.

#### a. Validasi Pertama

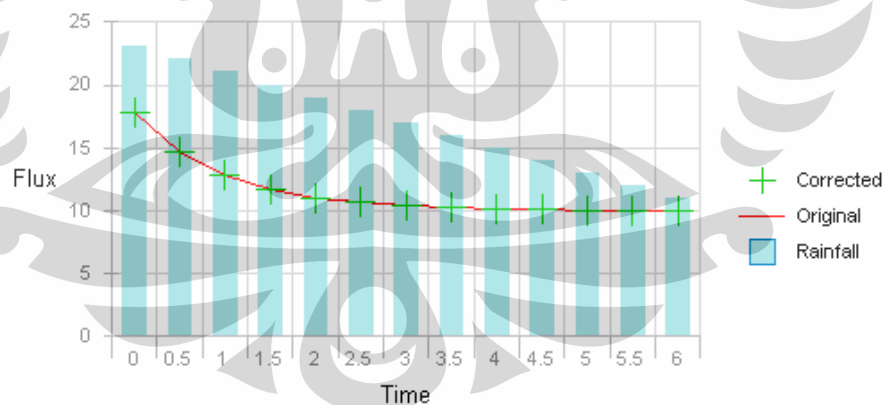
Validasi pertama dilakukan dengan melakukan pengujian subroutine horton saat semua intensitas hujan selalu lebih besar dari pada laju

infiltrasi tanah, hasil pengujian yang didapat ditunjukkan seperti dalam tabel berikut:

**Tabel L1a.** Hasil Pengujian Pertama

t (sec)	I (mm)	$f_{\text{original}}$ (mm/sec)	$f_{\text{Corrected}}$ (mm/sec)
0.0	23	17.79	17.79
0.5	22	14.72	14.74
1.0	21	12.87	12.88
1.5	20	11.74	11.75
2.0	19	11.05	11.06
2.5	18	10.64	10.64
3.0	17	10.39	10.39
3.5	16	10.24	10.24
4.0	15	10.14	10.14
4.5	14	10.09	10.09
5.0	13	10.05	10.05
5.5	12	10.03	10.03
6.0	11	10.02	10.02

Untuk memudahkan visualisasi perbandingan yang lebih jelas antara laju infiltrasi yang dihasilkan dari fungsi dengan laju infiltrasi yang dihasilkan oleh subroutine, dapat terlihat dalam gambar L1a.



**Grafik L1a.** Hasil Pengujian Pertama

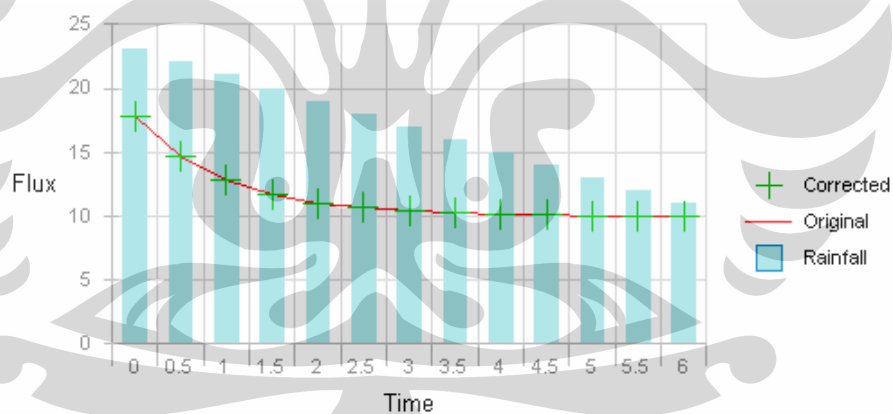
### **b. Validasi Kedua**

Validasi kedua dilakukan dengan melakukan pengujian subroutine horton saat semua intensitas hujan selalu lebih kecil dari pada laju infiltrasi tanah, hasil pengujian yang didapat ditunjukkan seperti dalam tabel berikut:

**Tabel L1b.** Hasil Pengujian Kedua

t (sec)	I (mm)	$f_{\text{original}}$ (mm/sec)	$f_{\text{Corrected}}$ (mm/sec)
0.0	3	17.79	17.79
0.5	7	14.72	17.21
1.0	9	12.87	15.97
1.5	10	11.74	14.56
2.0	9	11.05	13.27
2.5	8	10.64	12.34
3.0	7	10.39	11.71
3.5	6	10.24	11.27
4.0	5	10.14	10.98
4.5	4	10.09	10.78
5.0	3	10.05	10.65
5.5	2	10.03	10.57
6.0	1	10.02	10.52

Untuk memudahkan visualisasi perbandingan yang lebih jelas antara laju infiltrasi yang dihasilkan dari fungsi dengan laju infiltrasi yang dihasilkan oleh subroutine, dapat terlihat dalam gambar L1b.

**Grafik L1b.** Hasil Pengujian Kedua

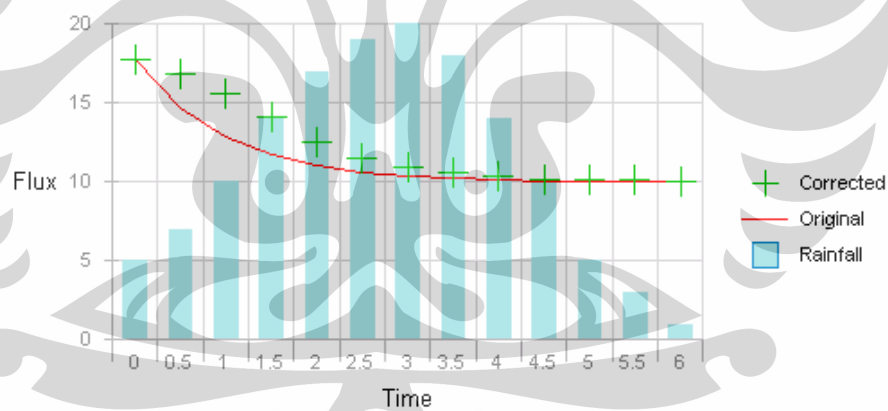
### c. Validasi Ketiga

Validasi ketiga dilakukan dengan melakukan pengujian subroutine horton saat semua intensitas hujan bervariasi terhadap pada laju infiltrasi tanah, hasil pengujian yang didapat ditunjukkan seperti dalam tabel berikut:

**Tabel L1c.** Hasil Pengujian Kedua

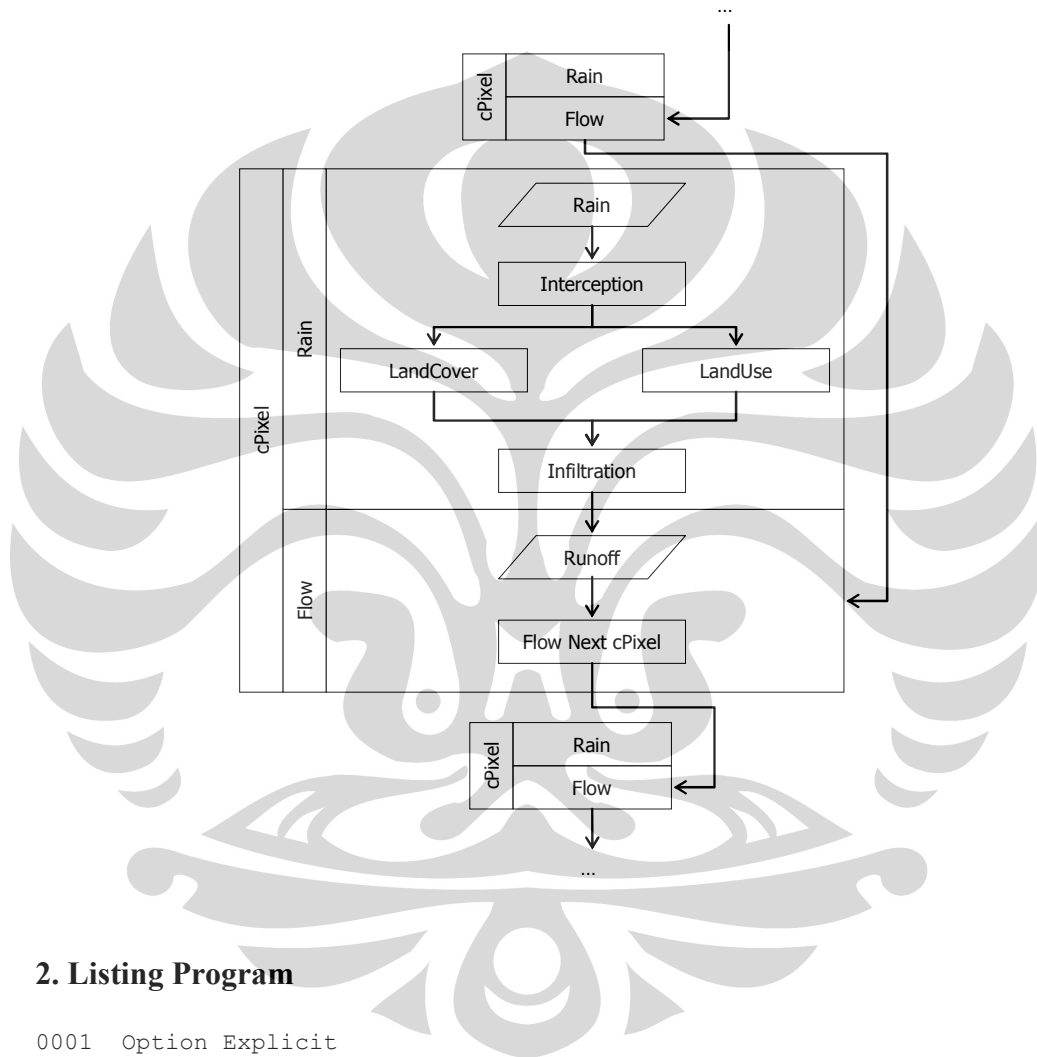
t (sec)	I (mm)	$f_{\text{original}}$ (mm/sec)	$f_{\text{Corrected}}$ (mm/sec)
0.0	5	17.79	17.79
0.5	7	14.72	16.85
1.0	10	12.87	15.64
1.5	14	11.74	14.14
2.0	17	11.05	12.53
2.5	19	10.64	11.54
3.0	20	10.39	10.93
3.5	18	10.24	10.57
4.0	14	10.14	10.34
4.5	10	10.09	10.21
5.0	5	10.05	10.13
5.5	3	10.03	10.10
6.0	1	10.02	10.09

Untuk memudahkan visualisasi perbandingan yang lebih jelas antara laju infiltrasi yang dihasilkan dari fungsi dengan laju infiltrasi yang dihasilkan oleh subroutine, dapat terlihat dalam gambar L1c.

**Grafik L1c.** Hasil Pengujian Kedua

## LAMPIRAN 3 KELAS CPIXEL

### 1. Struktur Simulasi Kelas



### 2. Listing Program

```

0001 Option Explicit
0002
0003 Public Enum eVegetalCover
0004     NoVegetation = 0
0005     LitteVegetation = 1
0006     MediumVegetation = 2
0007     DenseVegetation = 3
0008 End Enum
0009
0010 Public Enum eLandCover
0011     Asphalt = 0
0012     BrickPaving = 1
0013     RoofBuilding = 2
  
```

```

0014     GrassCovered = 3
0015 End Enum
0016
0017 Public Enum eLandUse
0018     UrbanBussiness = 0
0019     CommercialOffice = 1
0020     Residential_Homes = 2
0021     Residential_Condominium = 3
0022     Residential_Apartments = 4
0023     SuburbanResidential = 5
0024     Industrial_Light = 6
0025     Industial_Heavy = 7
0026     ParkOpenSpace = 8
0027     RailRoadWay = 9
0028 End Enum
0029
0030 Public Enum eSoilType
0031     Sandy_LitteVegetation = 0
0032     Sandy_DenseVegetation = 1
0033     Sandy_PoorDeveloped = 2
0034     Sandy_WellDeveloped = 3
0035     Loam_LitteVegetation = 4
0036     Loam_DenseVegetation = 5
0037     Loam_PoorDeveloped = 6
0038     Loam_WellDeveloped = 7
0039     Clay_LitteVegetation = 8
0040     Clay_DenseVegetation = 9
0041     Clay_PoorDeveloped = 10
0042     Clay_WellDeveloped = 11
0043 End Enum
0044
0045 Private Cli           As Double
0046 Private C             As Double
0047
0048 Private iCumul       As Double 'in mm
0049 Private iTime        As Double 'in sec
0050 Private f0           As Double 'in mm/sec
0051 Private fc           As Double 'in mm/sec
0052 Private k            As Double 'in 1/sec
0053
0054 Private Elev         As Double
0055 Private A            As Double 'in m^2
0056 Private Q            As Double 'in m^3/s
0057 Private Inflow      As Double 'in m^3/s
0058 Private NextPixel   As cPixel
0059
0060 Private Sub Class_Terminate()
0061     Set NextPixel = Nothing
0062 End Sub
0063
0064 Public Sub ResetSimulation()
0065     iCumul = 0
0066     iTime = 0
0067     Q = 0
0068 End Sub

```



```

0069
0070 Public Sub SimulateRain(ByVal I As Double, ByVal dt As Double)
0071     Dim f1 As Double
0072     Dim f2 As Double
0073
0074     '// Inflow From Near Cell
0075     Q = Inflw
0076     Inflw = 0
0077
0078     '// Interception
0079     I = I * Cli
0080
0081     '// LandUse and LandCover Effect
0082     I = I * (1 - C)
0083
0084     '// Runoff (LandUse and LandCover Effect)
0085     Q = Q + C * I / 1000 * A
0086
0087     '// Infiltration
0088     f1 = f(iTime)
0089     f2 = f(iTime + dt)
0090
0091     If f1 < I Then
0092         ' No Correction Needed
0093         iCumul = iCumul + (f1 + f2) / 2 * dt
0094
0095         '// Runoff (Infiltration Excess)
0096         Q = Q + (I - (f1 + f2) / 2) / 1000 * A
0097     Else
0098         ' Correction Needed
0099         If f2 < I Then
0100             ' Intersecting
0101             Dim mdt As Double
0102             Do While (I - f(iTime + mdt)) > 0.00001
0103                 mdt = mdt + (I - f(iTime + mdt)) / (-k * (f0 - f _
0104 fc) * Exp(iTime + mdt))
0105             Loop
0106
0107             ' Calculating Start to Mid
0108             iCumul = iCumul + I * mdt
0109             iTime = t(iCumul)
0110
0111             ' Calculating Mid To End
0112             f1 = f(iTime)
0113             f2 = f(iTime + (dt - mdt))
0114             iCumul = iCumul + (f1 + f2) / 2 * (dt - mdt)
0115
0116             '// Runoff (Infiltration Excess)
0117             Q = Q + (I - (f1 + f2) / 2) / 1000 * A
0118         Else
0119             ' No Intersecting
0120             iCumul = iCumul + I * dt
0121
0122             '// Runoff (Infiltration Excess)
0123             Q = Q + 0

```

```

0124         End If
0125     End If
0126
0127     ' Adjust Time
0128     iTime = t(iCumul)
0129 End Sub
0130
0131 Public Sub SimulateFlow()
0132     If Not IsNull(NextPixel) Then
0133         NextPixel.Inflow = NextPixel.Inflow + Q
0134     End If
0135     Q = 0
0136 End Sub
0137
0138
0139 '/////////////////////////////////////////////////////////////////
0140 '// Horton Infiltration Procedure and Funcions
0141
0142 ' Get Infiltration Rate by Time
0143 Private Function f(ByVal t As Double) As Double
0144     f = fc + (f0 - fc) * Exp(-k * t)
0145 End Function
0146
0147 ' Get Infiltration Cumulative by Time
0148 Private Function Cumul(ByVal t As Double) As Double
0149     Cumul = fc * t + (f0 - fc) / k * (1 - Exp(-k * t))
0150 End Function
0151
0152 ' Get Infiltration Time by Cumulative
0153 Private Function t(ByVal Cumulative As Double) As Double
0154     Do While Abs(Cumulative - Cumul(t)) > 0.00001
0155         t = t + (Cumulative - Cumul(t)) / f(t)
0156     Loop
0157 End Function
0158
0159
0160
0161 '/////////////////////////////////////////////////////////////////
0162 '// Class Properties
0163 Public Property Let Elevation(ByVal Value As Double)
0164     Elev = Value
0165 End Property
0166
0167 Public Property Get Elevation() As Double
0168     Elevation = Elev
0169 End Property
0170
0171 Public Property Let Area(ByVal Value As Double)
0172     A = Value
0173 End Property
0174
0175 Public Property Get Area() As Double
0176     Area = A
0177 End Property
0178

```

```

0179 Friend Property Let Runoff(ByVal Value As Double)
0180     Q = Value
0181 End Property
0182
0183 Public Property Get Runoff() As Double
0184     Runoff = Q
0185 End Property
0186
0187 Friend Property Let Inflow(ByVal Value As Double)
0188     Inflow = Value
0189 End Property
0190
0191 Friend Property Get Inflow() As Double
0192     Inflow = Inflow
0193 End Property
0194
0195 Public Property Let FlowTo(ByVal Value As cPixel)
0196     Set NextPixel = Value
0197 End Property
0198
0199 Public Property Get FlowTo() As cPixel
0200     Set FlowTo = NextPixel
0201 End Property
0202
0203 Public Property Let VegetalCover(ByVal Value As eVegetalCover)
0204     Select Case Value
0205         Case NoVegetation
0206             Cli = 1
0207         Case LitteVegetation
0208             Cli = 0.9
0209         Case MediumVegetation
0210             Cli = 0.7
0211         Case DenseVegetation
0212             Cli = 0.6
0213     End Select
0214 End Property
0215
0216 Public Property Let InterceptionCoef(ByVal Value As Double)
0217     Cli = Value
0218 End Property
0219
0220 Public Property Get InterceptionCoef() As Double
0221     InterceptionCoef = Cli
0222 End Property
0223
0224 Public Sub SetLandMark(ByVal Value1 As eLandUse, ByVal Value2 As _
0225 s eLandCover)
0226     Dim Clu           As Double
0227     Dim Clc           As Double
0228
0229     Select Case Value1
0230         Case UrbanBussiness
0231             Clu = 0.8
0232         Case CommercialOffice
0233             Clu = 0.6

```

```

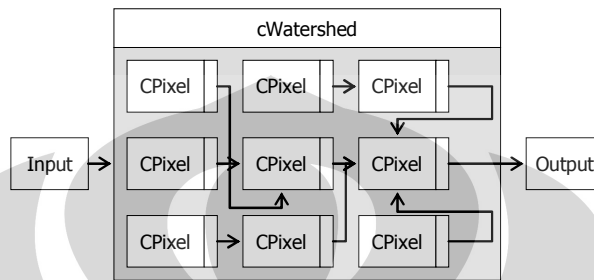
0234         Case Residential_Homes
0235             Clu = 0.4
0236         Case Residential_Condominium
0237             Clu = 0.5
0238         Case Residential_Apartments
0239             Clu = 0.7
0240         Case SuburbanResidential
0241             Clu = 0.3
0242         Case Industrial_Light
0243             Clu = 0.6
0244         Case Industrial_Heavy
0245             Clu = 0.8
0246         Case ParkOpenSpace
0247             Clu = 0.3
0248         Case_RailRoadWay
0249             Clu = 0.2
0250     End Select
0251
0252     Select Case Value2
0253         Case Asphalt
0254             Clc = 0.8
0255         Case BrickPaving
0256             Clc = 0.75
0257         Case RoofBuilding
0258             Clc = 0.85
0259         Case GrassCovered
0260             Clc = 0.1
0261     End Select
0262
0263     C = (Clu + Clc) / 2
0264 End Sub
0265
0266 Public Property Let LandEffectCoef(ByVal Value As Double)
0267     C = Value
0268 End Property
0269
0270 Public Property Get LandEffectCoef() As Double
0271     LandEffectCoef = C
0272 End Property
0273
0274 Public Property Let SoilType(ByVal Value As eSoilType)
0275     Select Case Value
0276         Case Sandy_LitteVegetation
0277             f0 = 0.0353: fc = 0.0032: k = 0.0055
0278         Case Sandy_DenseVegetation
0279             f0 = 0.0706: fc = 0.0036: k = 0.0052
0280         Case Sandy_PoorDeveloped
0281             f0 = 0.0172: fc = 0.0027: k = 0.0051
0282         Case Sandy_WellDeveloped
0283             f0 = 0.0233: fc = 0.0031: k = 0.0055
0284         Case Loam_LitteVegetation
0285             f0 = 0.0212: fc = 0.0008: k = 0.0107
0286         Case Loam_DenseVegetation
0287             f0 = 0.0422: fc = 0.0011: k = 0.0097
0288         Case Loam_PoorDeveloped

```

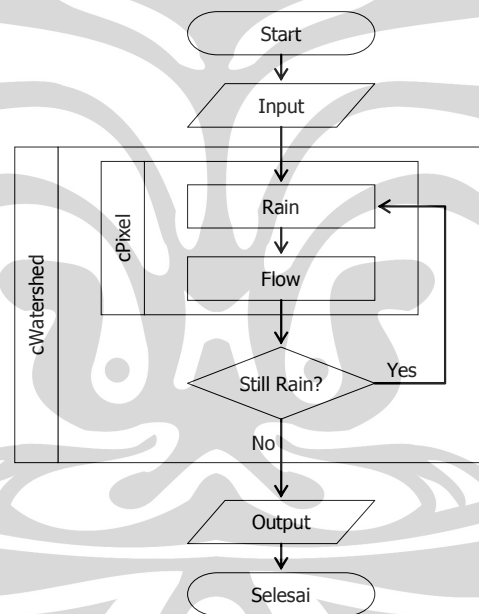
```
0289         f0 = 0.0117: fc = 0.0004: k = 0.0083
0290     Case Loam_WellDeveloped
0291         f0 = 0.0171: fc = 0.0007: k = 0.0107
0292     Case Clay_LitteVegetation
0293         f0 = 0.0071: fc = 0.0006: k = 0.0022
0294     Case Clay_DenseVegetation
0295         f0 = 0.0141: fc = 0.0008: k = 0.0157
0296     Case Clay_PoorDeveloped
0297         f0 = 0.0056: fc = 0.0002: k = 0.0017
0298     Case Clay_WellDeveloped
0299         f0 = 0.0098: fc = 0.0005: k = 0.0022
0300     End Select
0301 End Property
0302
0303 Public Sub SoilProperties(ByVal cf0 As Double, ByVal cfc As Doub _
0304 ble, ByVal ck As Double)
0305     f0 = cf0
0306     fc = cfc
0307     k = ck
0308 End Sub
0309
0310 Public Property Let Soil_f0(ByVal Value As Double)
0311     f0 = Value
0312 End Property
0313
0314 Public Property Get Soil_f0() As Double
0315     Soil_f0 = f0
0316 End Property
0317
0318 Public Property Let Soil_fc(ByVal Value As Double)
0319     fc = Value
0320 End Property
0321
0322 Public Property Get Soil_fc() As Double
0323     Soil_fc = fc
0324 End Property
0325
0326 Public Property Let Soil_k(ByVal Value As Double)
0327     k = Value
0328 End Property
0329
0330 Public Property Get Soil_k() As Double
0331     Soil_k = k
0332 End Property
```

## LAMPIRAN 4 KELAS CWATERSHED

### 1. Struktur Simulasi Kelas



Langkah 1. Delineasi Pixel dalam Watershed



Langkah 2. Simulasi Hujan - Alir

### 2. Listing Program

```

0001 Option Explicit
0002
0003 Private wWidth      As Long
0004 Private wHeight    As Long
0005
0006 Private wMapReady   As Boolean
0007 Private wMap()      As New cPixel
0008
0009 Private oX          As Long
0010 Private oY          As Long

```

```

0011
0012 Public Sub CreateWatershed(ByVal nWidth As Long, ByVal nHeight A _
0013 As Long)
0014     If wMapReady Then ResetData
0015
0016     wWidth = nWidth
0017     wHeight = nHeight
0018
0019     ReDim wMap(0 To ((nWidth * nHeight) - 1))
0020     wMapReady = True
0021 End Sub
0022
0023 Public Sub ResetSimulation()
0024     Dim n As Long
0025
0026     If wMapReady Then
0027         For n = 0 To UBound(wMap)
0028             wMap(n).ResetSimulation
0029         Next n
0030     End If
0031 End Sub
0032
0033 Public Sub Delineate()
0034     Dim x As Long, y As Long
0035     Dim MinElev As Double
0036     Dim Degree As Double
0037
0038     If wMapReady Then
0039         For y = 0 To (wHeight - 1)
0040             For x = 0 To (wWidth - 1)
0041                 MinElev = wMap(y * wWidth + x).Elevation
0042                 wMap(y * wWidth + x).FlowTo = Nothing
0043
0044                 '// Top Row
0045                 If y > 0 Then
0046                     If x > 0 Then
0047                         If wMap((y - 1) * wWidth + (x - 1)).Elev _
0048 vation < MinElev Then
0049                             MinElev = wMap((y - 1) * wWidth + (x _
0050 x - 1)).Elevation
0051                             wMap(y * wWidth + x).FlowTo = wMap(( _
0052 (y - 1) * wWidth + (x - 1))
0053                         End If
0054                     End If
0055
0056                     If wMap((y - 1) * wWidth + x).Elevation < Mi _
0057 inElev Then
0058                         MinElev = wMap((y - 1) * wWidth + x).Ele _
0059 evation
0060                         wMap(y * wWidth + x).FlowTo = wMap((y - _
0061 1) * wWidth + x)
0062                     End If
0063
0064                     If x < (wWidth - 1) Then
0065                         If wMap((y - 1) * wWidth + (x + 1)).Elev _

```

```

0066 vation < MinElev Then
0067                                     MinElev = wMap((y - 1) * wWidth + (x
0068 x + 1)).Elevation
0069                                     wMap(y * wWidth + x).FlowTo = wMap((
0070 (y - 1) * wWidth + (x + 1))
0071                                     End If
0072                                     End If
0073                                     End If
0074
0075                                     '// Middle Row
0076                                     If x > 0 Then
0077                                         If wMap(y * wWidth + (x - 1)).Elevation < Mi
0078 inElev Then
0079                                             MinElev = wMap(y * wWidth + (x - 1)).Ele
0080 evation
0081                                             wMap(y * wWidth + x).FlowTo = wMap(y * w
0082 wWidth + (x - 1))
0083                                         End If
0084                                     End If
0085
0086                                     If x < (wWidth - 1) Then
0087                                         If wMap(y * wWidth + (x + 1)).Elevation < Mi
0088 inElev Then
0089                                             MinElev = wMap(y * wWidth + (x + 1)).Ele
0090 evation
0091                                             wMap(y * wWidth + x).FlowTo = wMap(y * w
0092 wWidth + (x + 1))
0093                                         End If
0094                                     End If
0095
0096                                     '// Bottom Row
0097                                     If y < (wHeight - 1) Then
0098                                         If x > 0 Then
0099                                             If wMap((y + 1) * wWidth + (x - 1)).Elev
0100 vation < MinElev Then
0101                                                 MinElev = wMap((y + 1) * wWidth + (x
0102 x - 1)).Elevation
0103                                                 wMap(y * wWidth + x).FlowTo = wMap((
0104 (y + 1) * wWidth + (x - 1))
0105                                             End If
0106                                         End If
0107
0108                                         If wMap((y + 1) * wWidth + x).Elevation < Mi
0109 inElev Then
0110                                             MinElev = wMap((y + 1) * wWidth + x).Ele
0111 evation
0112                                             wMap(y * wWidth + x).FlowTo = wMap((y +
0113 1) * wWidth + x)
0114                                         End If
0115
0116                                         If x < (wWidth - 1) Then
0117                                             If wMap((y + 1) * wWidth + (x + 1)).Elev
0118 vation < MinElev Then
0119                                                 MinElev = wMap((y + 1) * wWidth + (x
0120 x + 1)).Elevation

```



```

0121             wMap(y * wWidth + x).FlowTo = wMap(( _
0122 (y + 1) * wWidth + (x + 1))
0123             End If
0124         End If
0125     End If
0126
0127     '// Error Tracing, Flow to pixel which direct to _
0128 o origin
0129     If (x - oX) = 0 Then
0130         Degree = 90
0131         If (oY - y) < 0 Then Degree = 270
0132     Else
0133         Degree = Atn((oY - y) / (x - oX)) / (22 / 7 _
0134 / 180)
0135         If (x - oX) < 0 Then Degree = Degree - 180
0136     End If
0137
0138     Select Case Degree
0139     Case 337.5 - 360, 0 - 22.5 'N
0140         If y > 0 Then
0141             wMap(y * wWidth + x).FlowTo = wMap(( _
0142 (y - 1) * wWidth + x)
0143             End If
0144
0145     Case 22.5 - 67.5 'NE
0146         If y > 0 And x < (wWidth - 1) Then
0147             wMap(y * wWidth + x).FlowTo = wMap(( _
0148 (y - 1) * wWidth + (x + 1))
0149             End If
0150
0151     Case 67.5 - 112.5 'E
0152         If x < (wWidth - 1) Then
0153             wMap(y * wWidth + x).FlowTo = wMap(y _
0154 y * wWidth + (x + 1))
0155             End If
0156
0157     Case 112.5 - 157.5 'EW
0158         If y < (wHeight - 1) And x < (wWidth - 1 _
0159 1) Then
0160             wMap(y * wWidth + x).FlowTo = wMap(( _
0161 (y + 1) * wWidth + (x + 1))
0162             End If
0163
0164     Case 157.5 - 202.5 'W
0165         If y < (wHeight - 1) Then
0166             wMap(y * wWidth + x).FlowTo = wMap(( _
0167 (y + 1) * wWidth + x)
0168             End If
0169
0170     Case 202.5 - 247.5 'WS
0171         If y < (wHeight - 1) And x > 0 Then
0172             wMap(y * wWidth + x).FlowTo = wMap(( _
0173 (y + 1) * wWidth + (x - 1))
0174             End If
0175

```

```

0176             Case 247.5 - 292.5 'S
0177                 If x > 0 Then
0178                     wMap(y * wWidth + x).FlowTo = wMap(y _
0179 y * wWidth + (x - 1))
0180                 End If
0181
0182             Case 292.5 - 337.5 'SN
0183                 If y > 0 And x > 0 Then
0184                     wMap(y * wWidth + x).FlowTo = wMap(( _
0185 (y - 1) * wWidth + (x - 1))
0186                 End If
0187             End Select
0188         Next x
0189
0190         If (y Mod 10) = 0 Then SetBusy "Delineating " & Int( _
0191 (y * 100 / wHeight) & "%")
0192     Next y
0193
0194     SetReady
0195 End If
0196 End Sub
0197
0198 Private Sub Class_Initialize()
0199     ResetData
0200 End Sub
0201
0202 Private Sub Class_Terminate()
0203     ResetData
0204 End Sub
0205
0206 Public Property Get PixelInfo(ByVal x As Long, ByVal y As Long) _
0207     As cPixel
0208     Set PixelInfo = wMap(y * wWidth + x)
0209 End Property
0210
0211 Public Property Get Width() As Long
0212     Width = wWidth
0213 End Property
0214
0215 Public Property Get Height() As Long
0216     Height = wHeight
0217 End Property
0218
0219 Public Property Get OriginX() As Long
0220     OriginX = oX
0221 End Property
0222
0223 Public Property Let OriginX(nVal As Long)
0224     oX = nVal
0225 End Property
0226
0227 Public Property Get OriginY() As Long
0228     OriginY = oY
0229 End Property
0230

```

```
0231 Public Property Let OriginY(nVal As Long)
0232     oY = nVal
0233 End Property
0234
0235 Private Sub ResetData()
0236     wWidth = 0
0237     wHeight = 0
0238
0239     Erase wMap()
0240     wMapReady = False
0241 End Sub
```



## LAMPIRAN 5

### KELAS PENDUKUNG LAIN

#### 1. kelas CRaster

```
0001 Option Explicit
0002
0003 Private rFileName As String
0004 Private rDisplay As New cBuffer
0005
0006 Private rWidth As Long
0007 Private rHeight As Long
0008 Private rSize As Double
0009
0010 Private rAnchorX As Double
0011 Private rAnchorY As Double
0012
0013 Private rNullValue As Double
0014 Private rMinValue As Double
0015 Private rMaxValue As Double
0016
0017 Private rRawReady As Boolean
0018 Private rRawData() As Double
0019
0020 Private Sub Class_Initialize()
0021     ResetData
0022 End Sub
0023
0024 Private Sub Class_Terminate()
0025     ResetData
0026 End Sub
0027
0028 Public Property Get Width() As Long
0029     Width = rWidth
0030 End Property
0031
0032 Public Property Get Height() As Long
0033     Height = rHeight
0034 End Property
0035
0036 Public Property Get AnchorX() As Double
0037     AnchorX = rAnchorX
0038 End Property
0039
0040 Public Property Get AnchorY() As Double
0041     AnchorY = rAnchorY
0042 End Property
0043
```