

BAB 2 LANDASAN KEPUSTAKAAN

Landasan kepustakaan ini adalah untuk menunjang pembuatan skripsi tentang metode *cluster* dan optimasi untuk pengelompokan biji wijen berdasarkan sifat warna cangkang biji dengan menggunakan metode *K-Means-ACO*.

2.1 Kajian Pustaka

Pada penelitian terdahulu, sudah pernah diteliti tentang pengelompokan biji wijen berdasarkan sifat warna cangkang bijinya. Penelitian pertama sendiri dilakukan oleh Adikadarsih (2015), dimana dalam penelitiannya adalah membandingkan metode pengelompokan dengan cara kuantitatif (metode pascale) dan kualitatif (metode manual). Metode kualitatif sendiri merupakan cara pengelompokan yang tidak efisien, karena membutuhkan waktu yang cukup lama dalam pengelompokan cangkang biji wijen tersebut. Pada penelitian sebelumnya yang hampir mirip dengan model pengelompokan Adikadarsih adalah Robbani (2017), Maulida (2017), dan Hanum (2017). Hasil dari pengujian pada penelitian tersebut adalah masing-masing nilai kekompakan dan nilai *fitnessnya* berbeda-beda. Sehingga pada penelitian skripsi ini bisa mengusulkan metode lain dengan menggunakan metode *Ant Colony Optimization* dan *K-Means*.

Maka pada kajian pustaka ini akan membahas referensi jurnal pada penelitian biji wijen sebelumnya, tentang metode *K-Means*, *Ant Colony Optimization* dan tentang metode *K-Means* dengan metode optimasi lainnya.

Penelitian pertama yang berjudul *Hyperspectral Image Clustering Using Ant Colony Optimization (ACO) Improved by K-Means Algorithm* oleh Sun Xu et al. Pada penelitian ini menggunakan data AVIRIS dengan dataset sekitar 200 dan memiliki 4 kelas. Dalam data ini membahas tentang gambar dengan skala 50 x 50 *pixels* pada suatu wilayah di Nevada, Amerika Serikat. Pada hasil yang didapat pada penelitian ini adalah jika memakai metode ACO saja mungkin hasilnya akan tidak bagus dari pada *K-Means-ACO*, karena algoritma ACO akan mencari jalan yang salah. *K-Means* sendiri akan membantu kerja ACO untuk tidak terlalu cepat dalam *update pheromone*. Dalam penggabungan metode pada penelitian ini ketika *K-Means-ACO* diuji dengan menggunakan *sum of the squared error (SSE)*, hasil dari metode *K-Means-ACO* lebih baik yaitu mendapatkan nilai SSE 370.7112 jika dibandingkan dengan algoritma *K-Means* yang mendapatkan nilai SSE sebesar 658.1526.

Pada penelitian jurnal kedua yang berjudul “Pengelompokan Biji Wijen Berdasarkan Sifat Warna Cangkang Biji dengan Menggunakan Metode IWOKM” oleh Ihwanudien Hasan Robbani menjelaskan tentang pengelompokan sifat warna cangkang biji wijen dengan menggunakan *Invasive Weed Optimization K-Means*. Hasil dari penelitian tersebut disimpulkan bahwa metode IWOKM dapat menjadi pengganti metode kuantitatif pada penelitian Adikadarsih (2015) dan terbukti lebih unggul dari pada metode *K-Means* biasa. Pengujian pada penelitian tersebut dengan membuat kelompok-kelompok data dan dari kelompok data tersebut diuji

nilai *fitness* dan nilai kekompakan kelompoknya. Pada perbandingan hasil pengelompokan *cluster* 1 dan 2 tidak jauh berbeda dibandingkan dengan pengelompokan pada penelitian terdahulu yaitu dengan metode IWOKM perbandingan antara C1 dan C2 adalah 52:233, sedangkan dengan metode kuantitatif perbandingan C1 dan C2 adalah 60:231.

Pada penelitian ketiga yang berjudul “Pengelompokan Biji Wijen Berdasarkan Sifat Warna Cangkang Biji dengan Menggunakan Metode PSO – *K-Means*” oleh Ardana Prakasita Devi yang membahas tentang pengelompokan biji wijen dengan metode PSO-*K-Means*, teknik dalam pengelompokan tersebut merupakan suatu teknik pada *data mining* untuk menentukan jumlah pengelompokan data kedalam beberapa kelompok. Metode *K-Means* sendiri sangat bergantung pada pusat titik *cluster* awal, sehingga solusi yang didapat sangat rentan pada daerah optimum lokal. Penelitian ini memakai metode PSO sebagai metode pendekatan untuk mengatasi masalah ini, metode PSO ini akan menemukan nilai titik pusat *cluster* yang optimum berdasarkan nilai *fitness* pada setiap partikelnya. Data yang digunakan pada penelitian ini sebanyak 291 data dengan atribut L^* , a^* , b^* . Hasil dari algoritma PSOKM ini menghasilkan pengelompokan lebih baik dibandingkan dengan algoritma *K-Means* saja.

Pada penelitian keempat yang berjudul “Pengelompokan Biji Wijen Berdasarkan Sifat Warna Cangkang Biji dengan Menggunakan Metode GA-*K-Means*” oleh Hanum Maulida yang membahas tentang pengelompokan biji wijen dengan metode GA-*K-Means*, pada penelitian ini menggunakan data berupa data kuantitatif dari hasil pengukuran warna cangkang biji wijen menggunakan *chromameter* yaitu terdiri dari 3 atribut yaitu L^* , a^* , b^* . Pada hasil pengujiannya metode GAKM lebih baik dibandingkan dengan metode *K-Means* saja, dari hasil *Silhouette Coefficient* metode GAKM memperoleh hasil 0,7701 sedangkan metode *K-Means* sendiri memperoleh hasil 0,7660. Dapat disimpulkan metode GAKM lebih baik dalam pengelompokan.

Metode *K-Means* dengan *Any Colony Optimization* adalah sebuah metode *hybrid*. Metode ini sangat bagus dalam pengelompokannya, dimana hasil dari pengelompokannya sangat baik. Penelitian ini akan menggunakan metode *K-Means-Ant Colony Optimization (K-Means-ACO)*. Maka diharapkan dengan menggunakan metode ini mendapatkan hasil yang optimal dalam pengelompokan biji wijen berdasarkan sifat warna cangkang biji secara kuantitatif.

Tabel 2.1 Kajian pustaka

No.	Judul	Object	Metode	Hasil
1.	<i>Hyperspectral Image Clustering Using Ant Colony Optimization (ACO) Improved</i>	Data AVIRIS dengan empat <i>sample</i> gambar	<i>K-Means</i> Using Ant Colony Optimization	Hasil dari metode <i>K-Means-ACO</i> lebih baik yaitu mendapatkan nilai SSE 370.7112 jika dibandingkan dengan algoritma <i>K-Means</i>

	<i>by K-Means Algorithm</i>			yang mendapatkan nilai SSE sebesar 658.1526.
2.	Pengelompokan Biji Wijen Berdasarkan Sifat Warna Cangkang Biji dengan Menggunakan Metode IWOKM	Data warna cangkang biji wijen dari hasil pengukuran alat <i>chromameter</i>	<i>Invasive Weed Optimization K-Means</i>	Hasil pada penelitian ini dengan menggunakan metode IWOKM nilai kekompakan kelompok jauh lebih baik dibandingkan dengan metode KM, dengan nilai rata-rata kekompakannya mencapai 0,7714.
3.	Pengelompokan Biji Wijen Berdasarkan Sifat Warna Cangkang Biji dengan Menggunakan Metode PSO – <i>K-Means</i>	Data warna cangkang biji wijen hasil pengukuran alat <i>chromameter</i>	<i>Particle Swarm Optimization K-Means</i>	Hasil dari algoritma PSO-KMeans ini menghasilkan pengelompokan lebih baik dibandingkan dengan algoritma <i>K-Means</i> saja.
4.	Pengelompokan Biji Wijen Berdasarkan Sifat Warna Cangkang Biji dengan Menggunakan Metode GA – <i>K-Means</i>	Data warna cangkang biji wijen hasil pengukuran alat <i>chromameter</i>	<i>Genetic Algorithm K-Means</i>	Hasil dari penelitian ini dengan metode GA-KM untuk pengelompokan biji wijen berdasarkan sifat warna cangkang biji menghasilkan anggota masing-masing kelompok 233 berbanding 58.
5.	Penerapan Metode <i>K-Means-ACO</i> untuk Pengelompokan Biji Wijen Berdasarkan Warna Cangkang Biji	Data warna cangkang biji wijen hasil pengukuran alat <i>chromameter</i>	<i>K-Means Ant Colony Optimization</i>	Hasil dari penelitian ini dengan menggunakan metode <i>K-Means-ACO</i> hasilnya lebih baik dari <i>K-Means</i> , dengan hasil nilai rata-rata <i>fitness</i> 10,1469 dan nilai

				kekompakan kelompok 0,7715.
--	--	--	--	-----------------------------

2.2 Wijen (*Sesamun Indicum L*)

Tanaman wijen (*Sesamun Indicum L.*) adalah sebuah tanaman dalam genus *indicum* dari keluarga *Pedaliaceae*, tanaman ini tumbuh dengan tinggi 50 hingga 100 cm. Sebagian besar dari tanaman tersebut digunakan untuk menghasilkan minyak nabati dan sisanya digunakan untuk keperluan pangan. Wijen memiliki beberapa varian warna, diantaranya yaitu berwarna kuning, merah, dan coklat (El Khier, 2008) . Dinegara berkembang tanaman wijen sendiri banyak dibudidayakan, pada tahun 1993 sendiri terdapat lahan sekitar 1000 hektar dari 7 hektar lahan, wijen pun tubuh dengan pesat di negara-negara berkembang. Wijen memiliki arti penting didalam dunia pertanian karena hasil dari produksi wijen sangat bagus dalam dunia pangan, wijen dapat tumbuh dengan baik di iklim tropis dan iklim sedang, kelembapan tanah, dan juga curah hujan yang cukup dapat memberikan hasil yang baik untuk kualitas wijen (Bennett, 1995).



Gambar 2.1 Tanaman dan biji wijen

Sumber: Tunde-Akintunde (2012)

2.2.1 Kegunaan Wijen

Tanaman wijen dapat diolah menjadi minyak ataupun bahan makanan. Berikut beberapa kegunaan wijen berdasarkan cara mengolahnya, diantaranya yaitu : (Feedipedia , 2016)

1. Biji wijen (*sesame seed*)

Fungsi utama penanaman wijen adalah untuk bahan makanan dan minyak. 65% biji wijen digunakan untuk proses ekstraksi minyak dan 35% untuk makanan. Biji wijen menghasilkan jumlah wijen dalam jumlah banyak dan setelah dimasak menciptakan rasa pedas. Sehingga, biji wijen sangat dibutuhkan di toko roti, industri permen dan makanan khas lainnya.

2. Minyak wijen (*sesame oil*)

Kegunaan utama biji wijen karena kandungan minyak yang tinggi. Biji wijen memiliki kandungan asam lemak tak jenuh ganda (*PUFA*) yang tinggi dan merupakan peringkat ke empat setelah safflower, kedelai dan jagung untuk kandungan *PUFA*. Minyak wijen mengandung sekitar 47% asam oleat dan 39% asam linoleat. Minyak wijen kaya akan tokoferol dan lignan (terutama sesamin dan sesamolin) yang memberikan stabilitas oksidatif yang luar biasa dibandingkan dengan minyak nabati lainnya. Ada dua jenis utama minyak wijen: yang pertama berwarna kuning pucat, memiliki bau seperti gandum dan rasa pedas dan digunakan untuk saus salad dan penggorengan dalam. Kedua, diperoleh dari biji panggang, berwarna kuning dan digunakan sebagai bahan penyedap rasa. Karena untuk stabilitasnya, minyak wijen dapat digunakan dalam produksi margarin dimana peralatan pendingin kurang. Minyak wijen digunakan dalam sediaan farmasi sebagai wahana pemberian obat, insektisida dan kosmetik.

3. Makanan minyak wijen (*sesame oil meal*)

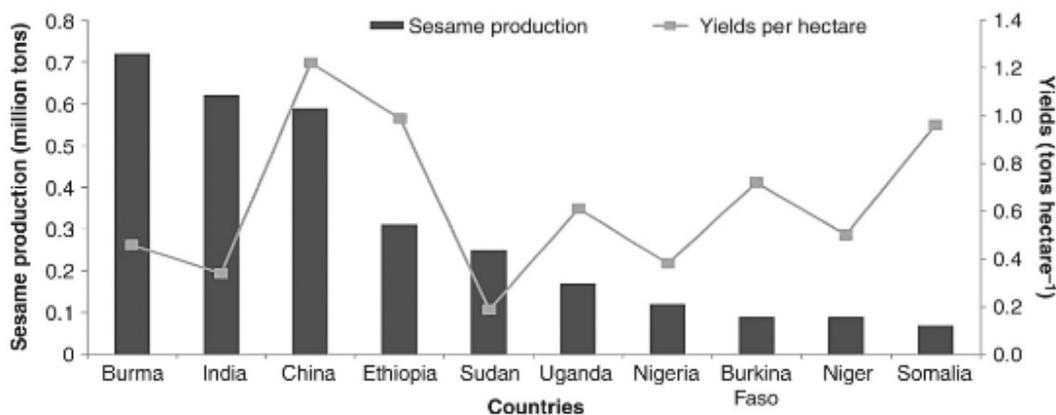
Makanan minyak wijen (atau kue minyak wijen) adalah produk yang kaya protein yang diperoleh setelah proses ekstraksi minyak. Berdasarkan proses ekstraksi makanan, minyak wijen dapat menjadi makanan bermutu (dari pengelupasan biji wijen) atau digunakan sebagai pakan ternak dan unggas (dari biji wijen yang tidak dipatenkan). Ini adalah sumber protein yang bermutu untuk hewan. Tidak seperti makanan minyak lainnya, minyak wijen biasanya diperoleh dengan ekstraksi mekanis saja (terkadang dengan ekstraksi mekanis yang diikuti oleh ekstraksi pelarut) dan kandungan sisa minyaknya cukup tinggi.

4. Kulit wijen (*sesame seed hulls*)

Cangkang atau kulit hasil dari pengelupasan biji wijen dapat digunakan sebagai makanan ternak dalam berkembang biak dan pemeliharaan unggas.

2.2.2 Produksi Wijen

Rata - rata produksi wijen dunia pada tahun 2010 adalah 3,84 juta ton yang tumbuh di lahan seluas 7,8 juta hektar. Produsen biji wijen terbesar di tahun 2013 adalah Burma. Eksportir biji wijen terbesar di dunia adalah India, dan Jepang merupakan pengimpor terbesar wijen. Di dunia, total budidaya wijen adalah 9.398.770 ha, menghasilkan 4,76 juta ton, yang meningkat dari 1,12 juta ton pada awal 1961-an. Negara-negara penghasil wijen utama di tahun 2007 adalah India, China, Burma (Myanmar), Sudan, Ethiopia, Uganda, dan Nigeria, sedangkan pada tahun 2001 produsen wijen terbesar adalah China dan India diikuti oleh Burma (4,2 juta ton) dan Sudan (3 Juta ton). Berikut adalah 10 negara dengan penghasil wijen terbesar didunia yang ditunjukkan pada Gambar 2.2. (Islam, 2016)



Gambar 2.2 Top 10 negara penghasil wijen di dunia

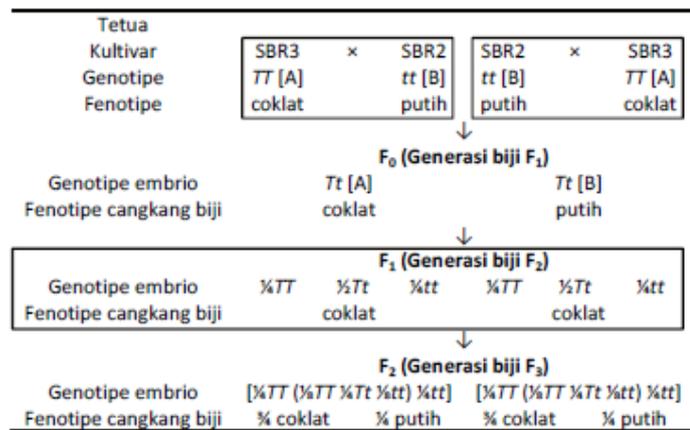
Sumber: Islam (2016)

2.3 Persilangan Wijen

Komponen dari tanaman wijen yang memiliki nilai ekonomi yang tinggi adalah warna biji wijen. Jika biji wijen tersebut bersih akan memiliki harga yang lebih tinggi, selain itu beberapa penelitian terdahulu dari warna biji wijen mempengaruhi hasil wijen, kadar minyak, dan kandungan biokimia pada wijen (Adikadarsih, 2015 dalam Robbani, 2017).

Persilangan wijen pada penelitian Adikadarsih (2015) menggunakan kultivar SBR 3 dan SBR 2. SBR 3 sebagai induk betina yang berwarna coklat bergenotipe TT. SBR 2 sebagai induk jantan yang berwarna putih bergenotipe tt. Hasil persilangan dua kultivar tersebut yaitu populasi bergenotipe Tt yang berwarna coklat seluruhnya. Sedangkan untuk respirknya, pada SBR 2 sebagai induk betina yang berwarna putih bergenotipe tt, lalu pada SBR 3 sebagai induk jantan yang berwarna putih seluruhnya bergenotipe TT. Hasil dari persilangan tersebut berwarna putih bergenotipe Tt. Sehingga dapat diketahui bahwa warna pada induk betina memiliki pengaruh paling besar pada hasil persilangan wijen. (Adikadarsih, 2015 dalam Robbani, 2017)

Dari hasil persilangan F₀, disilangkan Tetua Tt dengan warna coklat dan tetua Tt dengan warna putih, menghasilkan biji wijen berwarna coklat dengan genotipe TT : Tt : tt = 1 : 2 : 1. Lalu dilakukan persilangan hasil F₁, tetua Tt : Tt = coklat : putih, yang menghasilkan biji berwarna coklat dan putih dengan perbandingan 3:1. (Adikadarsih, 2015 dalam Robbani, 2017)

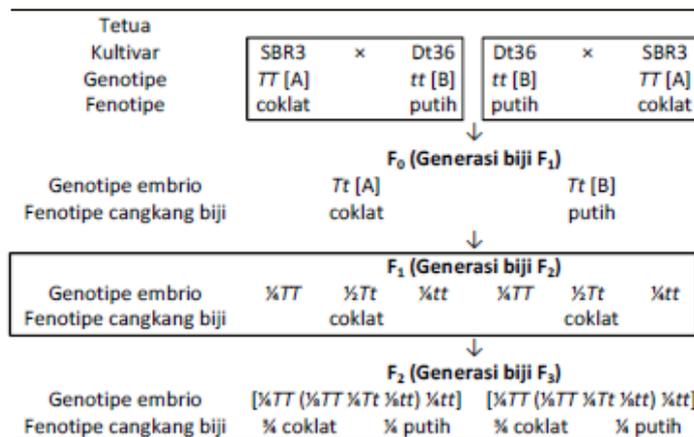


Gambar 2.3 Model alur pewarisan gen warna cangkang biji hasil persilangan SBR3 x SBR2 dan resiproknya

Sumber: Adikadarsih (2015) disitasi dalam Robbani (2017)

Persilangan serupa juga terjadi antara SBR3 dengan Dt36, SBR 3 sebagai induk betina yang berwarna coklat bergenotipe TT. Dt36 sebagai induk jantan yang berwarna putih bergenotipe tt. Hasil persilangan dua kultivar tersebut yaitu populasi bergenotipe Tt yang berwarna coklat seluruhnya. Sedangkan untuk resiproknya, pada Dt36 sebagai induk betina yang berwarna putih bergenotipe tt, lalu pada SBR 3 sebagai induk jantan yang berwarna coklat seluruhnya bergenotipe TT. Hasil dari persilangan tersebut berwarna putih bergenotipe Tt. Sehingga dapat diketahui bahwa warna pada induk betina memiliki pengaruh paling besar pada hasil persilangan wijen. (Adikadarsih, 2015 dalam Robbani, 2017)

Dari hasil persilangan F₀, disilangkan Tetua Tt dengan warna coklat dan tetua Tt dengan warna putih, menghasilkan biji wijen berwarna coklat dengan genotipe TT : Tt : tt = 1 : 2 : 1. Lalu dilakukan persilangan hasil F₁, tetua Tt : Tt = coklat : putih, yang menghasilkan biji berwarna coklat dan putih dengan perbandingan 3:1. (Adikadarsih, 2015 dalam Robbani, 2017)



Gambar 2.4 Model alur pewarisan gen warna cangkang biji hasil persilangan SBR3 x Dt36 dan resiproknya

Sumber: Adikadarsih (2015) disitasi dalam Robbani (2017)

Pada penelitian sebelumnya dilakukan perhitungan hasil pengelompokan warna secara kualitatif dan kuantitatif seperti pada Tabel 2.2. pada table tersebut menunjukkan bahwa ada perbedaan pada generasi F₂ (Adikadarsih, 2015, dalam Robbani, 2017).

Tabel 2.2 Hasil Pengelompokan biji wijen secara kuantitatif dan kualitatif

GENOTIF	KUANTITATIF			KUALITATIF		
	Coklat	Coklat muda	Putih	Coklat	Coklat muda	Putih
SBR 3	255			255		
SBR 2			23			23
Dt 36			185			185
F ₀ SBR3 x SBR2	3			3		
F ₀ SBR2 x SBR3			5			5
F ₀ SBR3 x Dt36	3			3		
F ₀ Dt36 x SBR3			4			4
F ₁ SBR3 x SBR2	57			57		
F ₁ SBR2 x SBR3	17			17		
F ₂ SBR3 x Dt36	11			11		
F ₁ Dt36 x SBR3	25			25		
F ₂ SBR3 x SBR2	216	15	60	210	27	54
F ₂ SBR2 x SBR3	201	35	92	196	38	94
F ₂ SBR3 x Dt36	308	76	38	298	82	42
F ₂ Dt36 x SBR3	92	25	29	90	23	33

Sumber : Adikadarsih (2015) disitasi dalam Robbani (2017)

2.4 Chromameter

Chromameter adalah alat untuk mengidentifikasi warna suatu objek atau benda. Alat ini menghasilkan *output* data warna dengan atribut L* a* b*. Penjelasan atribut warna L* a* b* akan dijelaskan pada subbab selanjutnya.

Chromameter dengan tipe Konica Minolta CR-410 telah dipakai dalam penelitian sebelumnya yang dilakukan oleh Adikadarsih (2015). Alat ini dapat mengevaluasi warna pada tekstur yang rata ataupun tidak. Pada pengujian terhadap alat ini, colorimeter terbukti handal dan memiliki akurasi tinggi dan

sangat efisien dalam membantu pengguna dalam menentukan kualitas warna, konsistensi, dan tampilan dari sampel objek tersebut. Alat ini berguna dalam banyak hal, diantaranya mengidentifikasi karakteristik warna pada sebuah objek, menentukan perbedaan warna antara objek dan memberi penilaian tentang LULUS/GAGAL dengan cepat apakah sudah sesuai dengan standar yang ditentukan. Dari beberapa keunggulan alat tersebut, *chromameter* dapat digunakan untuk mengidentifikasi warna pada makanan, material bangunan, aplikasi tekstil dalam control kualitas, jaminan kualitas / mutu dan bidang R&D. (Wibowo, n.d.)



Gambar 2.5 Konica Minolta Chroma Meter CR-410

Sumber : (Wibowo, n.d.)

2.5 Model CIELAB

CIELAB pertama kali dicetuskan pada tahun 1931 oleh Commission Internationale d'Eclairage (CIE) yang memiliki 3 *color* model yaitu L^* a^* b^* . Selanjutnya lebih dikenal dengan nama CIELAB. CIELAB dibuat sebagai ukuran yang pembentukan warnanya tidak mudah terpengaruh oleh sudut pandang *device* yang terkait, seperti (monitor, printer, komputer scanner, dsb) untuk menghasilkan *output (image)*. Hasil *output* model warna CIELAB menghasilkan atribut L^* , a^* , dan b^* , berikut penjelasannya (Gu, 2005) :

1. Nilai L^*

Yaitu antara 0-100, nilai 0 menunjukkan bahwa obyek sama sekali tidak memantulkan cahaya atau berwarna hitam dan jika bernilai 100 adalah sebaliknya berwarna putih.

2. Nilai a^*

Yaitu antara -100 hingga 100 yaitu menunjukkan derajat pigmentasi berwarna merah dan hijau, jika nilai a^* positif dan tinggi menunjukkan tingginya tingkat derajat pigmen berwarna merah dan sebaliknya.

3. Nilai b^*

Yaitu antara -100 hingga 100 nilai 100 menunjukkan derajat pigmentasi berwarna biru dan kuning, nilai b^* positif dan tinggi menunjukkan tingginya tingkat derajat pigmen berwarna kuning dan sebaliknya.

2.6 Clustering

Analisis *cluster* adalah suatu proses pembelajaran tanpa pengawasan (*unsupervised learning*) yang membagi satu set objek dengan memiliki kemiripan sama dalam satu *cluster* dan berbeda dengan *cluster* lainnya. Ada banyak algoritma pengelompokan yang telah digunakan di berbagai penelitian seperti pengenalan pola, *artificial intelligence*, teknologi informasi, pengolahan gambar, biologi, psikologi, dan pemasaran. Terkadang sangat sulit dalam mengidentifikasi algoritma yang sesuai dengan permasalahan yang ada. (Gan, 2007)

Tidak seperti pada klasifikasi, jika pada *clustering* pemberian label kelas tidak diketahui. Sehingga perlu ditemukan pengelompokan datanya yaitu dengan menggunakan data dalam jumlah banyak, sehingga jika menggunakan tenaga manusia kurang efektif dan efisien. Kemudian muncul strategi pengelompokan menggunakan metode *clustering* untuk mempartisi data. (Han, 2012)

Clustering adalah proses pengelompokan data ke dalam beberapa grup atau *cluster* sehingga objek dimasukkan ke dalam *cluster* yang memiliki kemiripan yang tinggi tetapi memiliki ketidakmiripan dengan objek pada *cluster* lain. Kemiripan dan ketidakmiripan dinilai berdasarkan nilai atribut yang melibatkan pengukuran jarak. *Clustering* sebagai perangkat data mining yang dapat digunakan untuk beberapa area seperti biologi, keamanan, bisnis, intelijen, dan pencarian web. (Han, 2012)

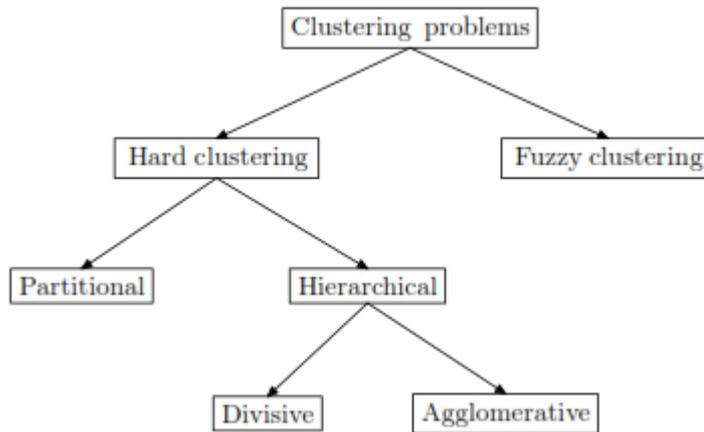
2.6.1 Teknik Clustering

Secara umum, algoritma pengelompokan dapat diklasifikasikan ke dalam dua kategori sebagai berikut : (Gan, 2007)

1. Algoritma Hirarki (*hierarchical algorithm*) terdiri dari dua jenis yaitu
 - *Divisive hierarchical algorithm* : hasil algoritmanya dari atas ke bawah yaitu algoritma dimulai dengan satu cluster besar yang berisi semua titik data dalam kumpulan data dan cluster selanjutnya.
 - *Agglomerative hierarchical algorithm* : hasil algoritma dari bawah ke atas yaitu algoritma dimulai dengan cluster masing-masing berisi titik data dan menggabungkan cluster.
2. Algoritma Partisi (*partitional algorithm*): algoritma partisi membuat data bertumpukan pada suatu partisi / kelompok tertentu.

Pada penelitian ini menggunakan *clustering* dengan teknik algoritma partisi (*partitional algorithm*). Algoritma partisi adalah teknik pengelompokan dengan cara membagi objek ke dalam kelompok tertentu, dimana setiap kelompok paling tidak memuat satu objek. Dapat dikatakan bahwa metode partisi hanya memiliki satu level partisi pada kumpulan data. Setiap objek paling tidak masuk ke dalam setiap kelompok. Sering kali, algoritma partisi dihitung berdasarkan jaraknya (*distance-based*). Algoritma partisi membuat suatu inisialisasi partisi, kemudian menggunakan teknik perubahan lokasi secara iteratif (*iterative relocation technique*) yaitu mencoba memperbaiki partisi dengan memindahkan objek dari satu grup ke lainnya. Kriteria umum dari teknik partisi yang baik adalah bahwa objek berada pada *cluster* yang sama atau berhubungan satu sama lain, sedangkan objek pada *cluster* sangat berbeda. Kebanyakan aplikasi menggunakan beberapa

metode heuristik populer seperti pendekatan *greedy* dengan algoritma *K-means* dan *K-medoid*. (Han, 2012)



Gambar 2.6 Diagram algoritma clustering

Sumber : Gan (2007)

2.7 K-Means (KM)

Algoritma konvensional atau *K-Means* merupakan salah satu algoritma yang paling sering digunakan untuk pengelompokan. Algoritma ini pertama kali dijelaskan oleh Macqueen (1967). Algoritma ini dirancang untuk *cluster* data numerik di mana setiap *cluster* memiliki pusat yang disebut *mean*. Algoritma *K-Means* diklasifikasikan sebagai metode pengelompokan *partitional* atau *nonhierarchical*. Pada algoritma ini, jumlah *cluster* k harus didefinisikan di awal. Adapun fungsi error algoritma *K-Means* yaitu memberikan inisialisasi k *cluster* dengan mengalokasikan data yang tersisa untuk *cluster* terdekat dan kemudian berulang kali mengubah keanggotaan kelompok menurut fungsi error sampai fungsi error tidak berubah secara signifikan atau keanggotaan dari kelompok tidak ada perubahan lagi. (Gan, 2007)

Algoritma *K-Means* konvensional menurut Hartigan (1975) akan di deskripsikan lebih jelas sebagai berikut : Misal D adalah sebuah data set dengan n objek dan C_1, C_2, \dots, C_k , k merupakan *cluster* terpisah dari D . Fungsi error dapat di definisikan sebagai berikut : (Gan, 2007)

$$E = \sum_{i=1}^k \sum_{p \in C_i} \sqrt{(x_1 - \mu(C_1))^2 + \dots + (x_n - \mu(C_n))^2} \quad (2.1)$$

$\mu(C_i)$: Centroid dari *cluster* C_i
 $d(x, \mu(C_i))^2$: jarak antara titik data x dan titik pusat $\mu(C_i)$

Secara ringkas, algoritma *K-Means* dapat dijelaskan sebagai berikut : (Gan, 2007)

Require : Data set D , jumlah *cluster* k , dimensi d

Berikut merupakan tahapan pada inisialisasi metode *K-Means*:

1. (C_1, C_2, \dots, C_k) = Inisialisasi pusat *cluster* secara acak diambil dari dataset D

Tahap Iterasi.

2. d_{ij} = jarak antara titik i ke *cluster* j (Persamaan 2.1);
 Hitung d_{ij} = jarak antara titik i ke *cluster* j menggunakan rumus *Euclidean Distance*.
3. $n_{ij} = \arg \min_{i \leq j \leq k} d_{ij}$;
 Tentukan nilai min jarak terdekat titik i ke *cluster* j .
4. Tetapkan titik i tergolong ke dalam *cluster* ke berapa n_i .
5. Hitung rata-rata centroid *cluster*.
6. Ulangi langkah 2 – 5 hingga tidak ada perubahan anggota *cluster*.
7. Hasil Output.

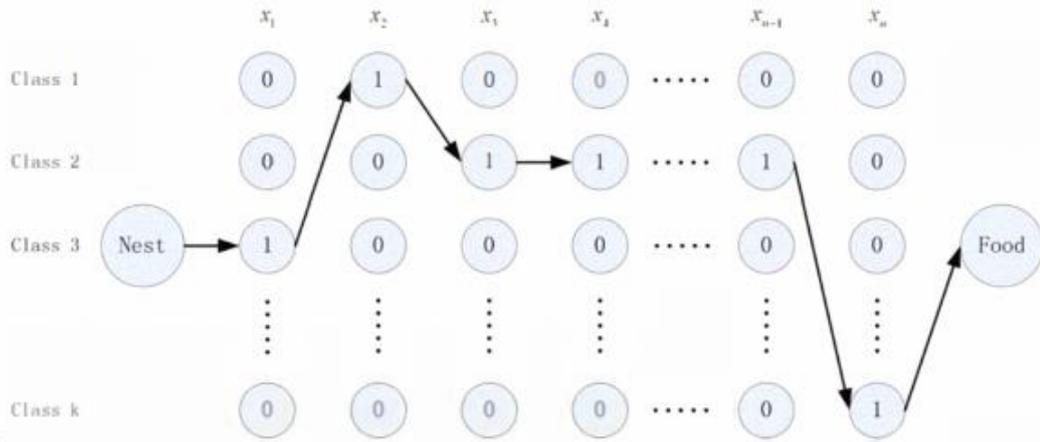
Algoritma *K-Means* ini memiliki dua tahap yaitu tahap inisialisasi dan tahap iterasi. Pada tahap inisialisasi, secara acak memilih *centroid* k *cluster* dari data set. Lalu pada tahap iterasi, menghitung jarak antara setiap objek dengan setiap *cluster* dan menentukan objek yang paling dekat dengan *cluster*.

2.8 Ant Colony Optimization (ACO)

Swarm intelligence adalah pendekatan yang relatif baru untuk pemecahan masalah dengan mengambil inspirasi dari perilaku serangga dan hewan lainnya. Secara khusus, semut telah memiliki cara yang paling baik dalam penerapannya yaitu dengan teknik optimasi yang dikenal sebagai optimasi koloni semut. Dari awal tahun sembilan puluhan, ketika pertama koloni semut optimasi algoritma diusulkan, ketertarikan beberapa peneliti terhadap metode ACO semakin meningkat dan banyak aplikasi yang sukses yang telah digunakan dengan metode tersebut (Gnanapriya, 2013).

Ant colony mempunyai beberapa karakteristik yang dianggap berguna. Dimana dia dapat melakukan tugas yang kompleks sebagai suatu grup tanpa perlu adanya koordinasi. *Ant-based techniques* yang pertama kali dilakukan adalah dengan menggunakan ACO, dimana digunakan oleh Dorigo et. al. (1996), untuk menyelesaikan *Traveling Salesman Problem* (TSP). Setiap semut yang berjalan akan meninggalkan *pheromone (stigmergic) signal* pada jalur yang dilaluinya. Jalur yang pendek akan meninggalkan sinyal yang lebih kuat. Semut berikutnya, pada saat memutuskan jalur mana yang harus dipilih, biasanya akan cenderung memilih untuk mengikuti jalur dengan sinyal yang paling kuat, sehingga jalur terpendek akan ditemui karena lebih banyak semut yang akan melewati jalur tersebut. Salah satu hal yang penting pada *ant-inspired technologies* adalah pada saat sistem telah menemukan solusi yang optimal, ACO akan dapat beradaptasi dengan cepat terhadap perubahan yang terjadi di sekitar. Adaptasi ini didasarkan pada *stigmergic* yang merupakan dasar dari *antsystem*. *Pheromone* yang lebih kuat akan dimiliki oleh solusi dengan jalur yang lebih optimal pada akhir suatu algoritma. (Santosa, 2010)

Berikut dibawah ini adalah contoh solusi hasil jarak terpendek yang berhasil dilewati semut ditunjukkan pada Gambar 2.7.



Gambar 2.7 Solusi jarak optimal Ant Colony Optimization (ACO)

Sumber: Xu (2010)

Untuk menentukan nilai probabilitas digunakan untuk semut mencari jalan/path terbaik. Hasil yang didapat dari nilai probabilitas yang tertinggi akan diproses untuk *update pheromone*. Jika semut menempati sebuah *path* yang terbaik, maka nilai probabilitas tersebut akan bernilai/mendekati 1. Dan jika semut tidak menempati *path* terbaik maka nilai probabilitas bernilai/mendekati 0.

Metode ACO sendiri bekerja dengan mencari solusi optimal, berikut merupakan tahap metode ACO :

1. Menghitung probabilitas semut *i* dalam *path j*.

Menghitung probabilitas di dapatkan dari konsentrasi *pheromone* . Inisialisasi konsentrasi *pheromone* pada semua node adalah sama dan akan di perbarui setelah semut memilih *path*. Berikut ini proses menghitung probabilitas semut:

$$p_{ij} = \begin{cases} \frac{\tau_{ij}}{\sum_{j=1}^k \tau} & x_i \notin X' \\ 1 & x_i \in X' \text{ and } j = \arg \max_j \{\tau_{ij}\} \\ 0 & x_i \in X' \text{ and } j \neq \arg \max_j \{\tau_{ij}\} \end{cases} \quad (2.2)$$

Keterangan :

P_{ij} = Probabilitas pada semut ke *i* *path j*

τ_{ij} = *Pheromone* konsentrasi pada semut ke *i* *path j*

2. Penambahan dan Penguapan *Pheromone (Update Pheromone)*.

Semut akan meninggalkan *pheromone* pada *path j*. Proses *update pheromone* dibagi ke dalam 3 jenis yaitu *ant-density*, *ant-quantity* dan *ant-cycle*, dari ketiganya yang paling sering digunakan adalah *ant-cycle*. Pada *ant cycle* mengasumsikan ada beberapa semut yang mencari jalan secara serentak dalam setiap iterasi dan hanya memperbarui *pheromone* dari titik di *path* terbaik pada iterasi terakhir. Dalam melakukan proses *update pheromone* memiliki dua bagian yaitu penguapan

secara alami (*pheromone* menurun) dan pelepasan oleh semut (*pheromone* naik). Semua titik *pheromone* pasti menguap namun hanya titik pada *path* terbaik yang dapat dilepaskan oleh semut. Semakin baik jalan yang dilalui, semakin banyak *pheromone* yang didapatkan dan akan dilalui oleh semut lainnya. *Pheromone* pada *path* j akan dihitung dengan rumus sebagai berikut:

$$\tau_{ij}^{new} = \begin{cases} \rho\tau_{ij}^{old} + Q/f_{min} & y_{ij} \text{ adalah path terbaik} \\ \rho\tau_{ij}^{old} & y_{ij} \text{ adalah bukan path terbaik} \end{cases} \quad (2.3)$$

Keterangan :

τ_{ij} = *Pheromone* baru

ρ = Koefisien penguapan

τ_{ij}^{old} = Inisialisasi *pheromone*

Q = Kecepatan *pheromone*

f_{min} = Fungsi objektif optimal (nilai *fitness* terkecil)

Dalam $\rho \in (0,1)$ merupakan koefisien penguapan yang berarti kecepatan dalam penguapan *pheromone*, f_{min} merupakan fungsi objektif optimal pada iterasi saat itu, fungsi objektif dalam penelitian ini menggunakan nilai *fitness* yang optimal, Q adalah bilangan positif bernilai konstan yang berarti kecepatan dalam melakukan *update pheromone*. Jika Q bernilai besar maka akan cepat konvergen dan mudah jatuh pada area *local optimum*, sedangkan jika Q bernilai kecil maka berlaku kebalikannya. Setelah proses *update pheromone* maka memulai iterasi baru.

2.9 K-Means - Ant Colony Optimization (K-Means-ACO)

Dalam algoritma *K-Means-ACO*, proses iterasi dilakukan oleh metode ACO sedangkan metode *K-means* hanya melakukan pengelompokan data tanpa memperbarui pusat centroid *K-means*. Pada hasil dari algoritma *K-Means* pengelompokannya diklasifikasikan dengan menggunakan prinsip jarak minimal.

Algoritma ACO susah konvergen untuk mendapatkan solusi optimal pada area yang besar. Tanpa elitisitas informasi, algoritma ACO hanya menggunakan *pheromone*. Jika kecepatan penggunaan *pheromone* cepat ($\rho =$ kecil dan $Q =$ besar) maka akan mudah jatuh pada lokal optimum. Sedangkan jika kecepatan penggunaan *pheromone* lambat ($\rho =$ besar dan $Q =$ kecil) maka jumlah iterasi meningkat dan akan membuat efisiensi algoritma melambat. (Xu, 2010)

Sehingga untuk memperbaiki efisiensi algoritma dan memastikan akurasinya, memungkinkan untuk meningkatkan jumlah *pheromone* yang mana akan mengubah strategi dari semut dalam pencarian *path*. Pertama algoritma *k-means* digunakan untuk mendapatkan hasil pengelompokan data dan mengambil hasil elitisitas informasi dari ACO yang digunakan untuk inisialisasi posisi *pheromone*, yang mana dapat menghasilkan jumlah *pheromone* dari setiap titik berhubungan dengan hasil dari *K-means*. (Xu, 2010)

Algoritma *K-Means-ACO* adalah sebuah metode untuk pencarian solusi optimal pada area yang besar seperti ditunjukkan pada Gambar 2.7. Semut berjalan dari sarang menuju makanan dan disambungkan dengan titik (*nodes*) yang bernilai 1, sehingga hal ini dapat digunakan untuk semut dalam menentukan *path* yang akan dilalui oleh semut. Jalan terpendek dari sarang menuju makanan akan ditemukan oleh semut sebagai solusi optimal dari hasil pengelompokan.

Berikut merupakan secara singkat tahapan pada algoritma *K-Means-ACO* :

1. Inisialisasi Posisi Awal Koloni Semut

Inisialisasi posisi koloni semut dilakukan beberapa semut untuk melewati satu tempat berdasarkan hasil inialisasi parameter awal, maka dalam satu *path*/tempat terdapat 3 semut. Dalam menentukan titik pusat secara *random* dilakukan dengan menggunakan fungsi *RAND*, seperti yang dijelaskan pada Persamaan 2.4.

$$x_i^j = x_{min}^j + \mathit{rand}(0, 1)(x_{max}^j - x_{min}^j) \quad (2.4)$$

Keterangan :

x_i^j = data *random cluster* j, data i

x_{min}^j = batas nilai minimal

x_{max}^j = batas nilai maksimal

2. Menentukan Keanggotaan Data dengan *K-Means*.

K-Means merupakan algoritma yang paling sering digunakan untuk metode pengelompokan. Algoritma *K-Means* sendiri ditemukan oleh Macqueen (1967). Algoritma ini dirancang untuk melakukan *cluster* data, diaman *cluster* pusat yang disebut *mean*. Perhitungan jarak dalam masing-masing data dengan seluruh titik pusat yang ada dilakukan dengan cara menggunakan rumus *Euclidean Distance*. Dalam penentuan keanggotaan kelompok dimana jarak data dan titik pusat kelompok dicari dengan jarak terpendek. Berikut merupakan tahapan algoritma *K-Means* ditunjukkan pada Persamaan 2.1.

3. Menghitung Nilai Rata-Rata *Fitness*

Untuk tahapan ini akan dihitung nilai *fitness*nya dari kelompok data yang sudah terbentuk, perhitungannya sendiri dilakukan dengan jumlah data keseluruhan pada C1 dan C2. Berikut merupakan perhitungan nilai rata-rata *fitness* ditunjukkan pada Persamaan 2.5.

$$\mathit{fitness}(x, y) = \frac{1}{M} \sum_{j=i}^N \left(\frac{1}{N} \sum_{j=1}^N (x_j - y_j)^2 \right) \quad (2.5)$$

Dalam persaman diatas dapat dijelaskan :

x = Nilai x ke indeks j

y = Nilai y ke indeks j

M = Jumlah titik pusat *cluster*

N = jumlah data

4. Menghitung Nilai Probabilitas

Tahap ini akan dihitung nilai probabilitas setiap semut dalam melewati sebuah *path*, nilai probabilitas ini digunakan pada semut untuk saling berkomunikasi. Jika nilai probabilitas ini dikatakan baik maka semut tersebut akan melewati *path* terbaiknya, sebaliknya jika semut tidak melewati *path* terbaik, maka tidak dihitung nilai probabilitasnya. Berikut merupakan rumus untuk menentukan nilai probabilitas ditunjukkan pada Persamaan 2.2.

5. Menghitung *Update Pheromone*

Siklus semut dapat diasumsikan bahwa terdapat beberapa semut yang akan mencari jalan secara bersamaan pada setiap iterasi, dan hanya akan memperbarui *pheromone* dari node yang memiliki jalan terbaik dalam sebuah iterasi saat ini. Ketika semut melewati jalur terbaiknya maka semut tersebut akan melepaskan *pheromonenya*. Berikut merupakan rumus perhitungan untuk menentukan *update pheromone* ditunjukkan pada Persamaan 2.3.

6. Menghitung *Update Posisi*

Semut melakukan *update* posisi jika dalam penemuan posisi terbaik belum mengalami konvergen. Dalam perhitungan *update* posisi terdapat dua kondisi yaitu jika semut pada posisi terbaik maka posisi semut tidak berubah dan jika bukan pada posisi terbaik maka dilakukan proses *update* posisi.

Proses *update* posisi mengacu pada nilai *update pheromone* terbaik yang mendekati nilai 1. Jika nilai *update pheromone* pada iterasi sebelumnya sudah didapatkan maka akan mengambil nilai mean pada hasil pengelompokan C1 dan C2. Nilai Mean untuk menghitung *update* posisi semut jika semut memiliki nilai feromon tertinggi atau pada posisi terbaik.

Berikut merupakan rumus menghitung *update* posisi menggunakan nilai *RANDBETWEEN* yang ditunjukkan pada Persamaan 2.6.

$$x = \text{RANDBETWEEN}((\text{Mean} - \text{Std.Dev}) * 10000; (\text{Mean} + \text{Std.Dev}) * 10000/10000) \quad (2.6)$$

2.10 *Silhouette Coefficient*

Silhouette coefficient adalah suatu metode untuk digunakan sebagai pengujian dalam kekompakan kelompok pada suatu data. Metode ini akan dilihat dikualitas pada suatu kelompok data. Pada nilai metode *silhouette* akan merepresentasikan apakah objek pada suatu kelompok mempunyai kemiripan jika dibandingkan dengan kelompok yang lain. *Silhouette* memiliki suatu nilai untuk mengukur kualitas pada kekompakan kelompok tersebut, yaitu nilai berkisar antara -1 sampai dengan 1. Jika pada nilai tersebut semakin tinggi (mendekati 1) maka pada kelompok tersebut memiliki suatu kemiripan. Pada proses pengelompokan dikatakan sesuai, jika pada suatu titik memiliki nilai yang tinggi dan jika pada titik tersebut memiliki nilai yang rendah maka pengelompokan

masih belum sesuai. Metode *silhouette* dapat dihitung dengan menggunakan *Euclidean distance*, dan *Manhattan*. (Rousseeuw, 1987)

Berikut merupakan langkah-langkah pada perhitungan metode *silhouette*: (Han, 2012)

- a. Hitung jarak rata-rata data ke - o dengan semua data yang berada dalam satu kelompok.

$$a(o) = \frac{\sum_{o' \in C_i, o' \neq o} \text{dist}(o, o')}{|C_i| - 1} \quad (2.8)$$

Keterangan:

o : Data ke-o pada kelompok ke-i

o' : Data lain pada kelompok i selain data ke-o

C_i : Kelompok ke-i

|C_i| : Jumlah data pada kelompok ke-i

dist(o, o') : Jarak data o dengan data o'

- b. Hitung jarak rata-rata jarak data ke - o dengan semua data yang berada pada kelompok lain kemudian diambil nilai paling minimum

$$b(o) = \min_{C_j : 1 \leq j \leq k, j \neq i} \left\{ \frac{\sum_{o' \in C_j, o' \neq o} \text{dist}(o, o')}{|C_j|} \right\} \quad (2.9)$$

- c. Hitung nilai *Silhouette*.

$$s(o) = \frac{b(o) - a(o)}{\max\{a(o), b(o)\}} \quad (2.10)$$