INTERPOLASI GERAKAN ROTASI ANIMASI 3D MENGGUNAKAN BILANGAN QUARTENION DAN IMPLEMENTASINYA PADA FUNGSI SLERP *(SPHERICAL LINEAR INTERPOLATION)*

Heru Arwoko [1], Endah Asmawati [2], Susana Limanto [3]

[1),2),3)] Fakultas Teknik - Jurusan Informatika - Universitas Surabaya
Jl. Raya Kalirungkut No.1 Surabaya, Indonesia
Email: heruarwoko@yahoo.co.id

Abstrak. *Interpolasi animasi obyek 3D memiliki peranan penting pada pembuatan film 3D. Pada animasi gerakan memutar dapat digunakan aplikasi bilangan quaternion untuk menghasilkan interpolasi yang bagus. Rotasi vektor menggunakan bilangan quaternion berhasil diterapkan pada saat interpolasi rotasi obyek dari satu keyframe ke keyframe yang lain. Interpolasi quartenion pada fungsi SLERP, spherical linear interpolation memiliki hasil interpolasi yang bagus. Sedangkan fungsi lainnya yaitu LERP, linear interpolation menghasilkan interpolasi rotasi yang lebih jelek. Pada artikel ini dijelaskan mendetail fungsi SLERP yang sangat menguntungkan pada pembuatan animasi 3D saat gerakan memutar dibandingkan hasil fungsi LERP.*

Kata kunci : *interpolasi, animasi, quaternion, rotasi vektor, SLERP, LERP.*

1. Pendahuluan

Dewasa dunia hiburan animasi 3D sedang berkembang maju. Hal ini tidak lepas dari teknik pembuatan animasi yang canggih dengan menerapkan konsep interpolasi obyek dari keyframe satu ke keyframe yang lain. Tanpa interpolasi tidak dapat dihasilkan hasil animasi yang halus apalagi jika pembentukan keyframe animasi dilakukan secara manual, tentu hasilnya akan memakan waktu lama dan hasilnya jelek tidak mendekati gerakan realitas. Banyak film animasi telah diproduksi contohnya Pixar, Disney, Paramount Animation, dan studio-studio lainnya. Pada film animasi tersebut tidak lepas dari gerakan obyek 3D yang tidak lain merupakan bentuk transformasi koordinat 3D meliputi translasi (pergeseran), skala, dan rotasi obyek. Khusus transformasi jenis rotasi perlu menggunakan aplikasi vektor quartenion, karena vektor quartenion dapat menyimpan besaran rotasi secara tepat. Pada artikel ini dibahas keunggulan interpolasi SLERP pada quartenion jika dibandingkan menggunakan LERP. Dahulu membuat film animasi sangat susah karena dilakukan secara tradisional. Untuk membuat suatu karakter bergerak diperlukan beratus-ratus frame. Belum lagi jika film tersebut memiliki warna. Proses pewarnaan dilakukan perlembar frame. Setelah tiap frame diwarnai versi final dari frame-frame tersebut dilanjutkan ke *celluloid* untuk diproduksi ke produk finalnya. Maka dari itu tidak dipungkuri lagi bahwa kemajuan bidang animasi adalah karena ditemukannya *computer animation*. Dengan adanya *computer animation,* perpindahan gerak pada sebuah karakter tidak perlu lagi digambar pada ratusan frame namun dapat dengan mudah dilakukan dengan interpolasi linear sederhana dari keyframe satu ke keyrame yang lain. Selain itu penerapan rumus fisika kinematika dan dinamika menjadikan hasil interpolasi gerakan animasi 3D tampak lebih natural sesuai dengan hukum alam. Mulanya perubahan pada objek 3D dinamis sangat sulit didapatkan namun dengan adanya *computer animation*, perubahan bentuk sebuah objek dapat diperkirakan secara otomatis. Contohnya seperti bola terpental yang ketika menyentuh tanah bola tersebut berbentuk oval dan ketika mencapai titik tertingginya bola tersebut berbalik membentuk sebuah lingkaran sempurna. Lingkup dari *computer animation* sebenarnya sangat banyak, tidak hanya mengatur pergerakan sebuah gambar namun juga mengatur pencahayaan, sudut kamera, pergerakan kamera, bayangan, suara, dan warna.

2. Pembahasan

Interpolasi adalah cara menentukan nilai yang berada di antara dua nilai diketahui berdasarkan suatu fungsi persamaan. Fungsi yang digunakan untuk mengatur pergerakan interpolasi sebuah karakter pada *computer animation* untuk gerakan memutar adalah SLERP dan LERP. Fungsi SLERP atau spherical linear interpolation adalah implementasi dari interpolasi bilangan quartenion. Sebenarnya cara merotasikan sebuah karakter memiliki banyak alternatif yaitu menggunakan sudut Euler, rotasi matriks, ataupun rotasi menggunakan bilangan quartenion SLERP dan LERP. Pada artikel ini dibahas keunggulan SLERP pada interpolasi obyek 3D untuk gerakan memutar dibandingkan menggunakan LERP.

2.1. Bilangan Quartenion

Bilangan quartenion mulai diperkenalkan oleh Sir William Rowan Hamilton pada tahun 1843. Bilangan quartenion merupakan jawaban para ilmuwan yang mencoba mencari ekivalen tiga dimensional dari bilangan kompleks. Quartenion didefinisikan dalam bentuk [1] :

$$z = a + ib + jc + kd \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

Objek ini disebut dengan '*quartenion*', merupakan penjumlahan aljabar dari bagian riil (nilai a) dan bagian imajinernya (nilai b, c , d) dinamai '*vektor*'.

Perkalian Quartenion

Jika ada dua quartenion q_1 dan q_2, sebagai berikut :

$$q_1 = s_1 + ix_1 + jy_1 + kz_1$$
$$q_2 = s_2 + ix_2 + jy_2 + kz_2$$

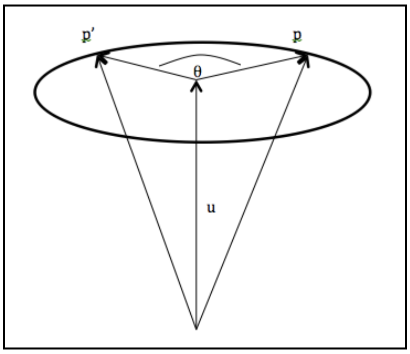
maka perkalian kedua quartenion adalah :

$$q_1 q_2 = s_1 s_2 - v_1 \cdot v_2 + s_1 v_2 + s_2 v_1 + v_1 \times v_2 \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2)$$

2.2. Rotasi Vektor Menggunakan Bilangan Quartenion

Rotasi vektor merupakan salah satu aplikasi dari bilangan quaternion. Suatu vektor p akan dirotasikan sejauh sudut θ pada sumbu axis yang diberikan oleh unit vektor u. Vektor p yang telah dirotasikan melalui sumbu u menghasilkan vektor baru p' didapat dengan operasi seperti berikut [2] :

$$p' = qpq^{-1}$$



Gambar 1. Rotasi vektor menggunakan quaternion

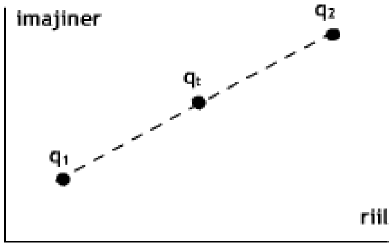
Ilustrasi dari rotasi vektor dapat dilihat pada gambar 1.

Dimana penjabaran tiap variabelnya adalah seperti di bawah ini:

$$p = xi + yj + zk$$
$$q = \cos(\frac{\theta}{2}) + \sin(\frac{\theta}{2})u$$
$$q^{-1} = \cos(\frac{\theta}{2}) - \sin(\frac{\theta}{2})u$$
$$i^2 = j^2 = k^2 = -1$$
$$ijk = -1; ij = k; jk = i; ki = j$$

2.3. LERP (Linear Interpolation)

Lerp atau Interpolasi linear adalah cara menentukan nilai yang berada di antara dua nilai diketahui berdasarkan persamaan linear (persamaan garis lurus). Persamaan linear disebut juga persamaan garis lurus karena jika hasil persamaan linear digambarkan pada kertas grafik maka bentuk kurvanya adalah garis lurus, seperti tampak pada gambar 2.



Gambar 2. Interpolasi linier quartenion q_t dari quartenion q_1 dan q_2

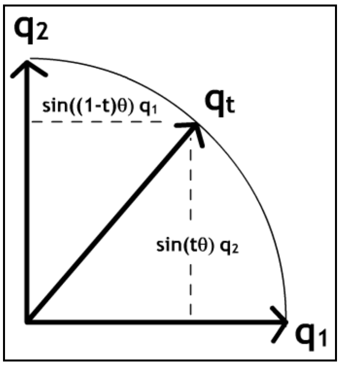
Metode standar untuk menginterpolasi sebuah titik antara dua orientasi vektor secara lebih halus adalah menggunakan persamaan linier. Misalkan orientasi pertama adalah q_1 dan orientasi kedua adalah q_2. Titik yang diinterpolasi direpresentasikan dengan q_t. Parameter interpolasi t akan menginterpolasi dari q_1 saat t = 0 sampai q_2 saat t = 1. Maka, rumus standar interpolasi linier [4] adalah

$$q_t = q_1 + t(q_2 - q_1) \qquad \text{..(3)}$$

2.4 SLERP (Spherical Linear Interpolation)

Hasil interpolasi obyek quartenion akan lebih baik jika ditinjau dari pendekatan vektor geometrinya, Slerp melakukan *spherical interpolation* secara vektor dari kedua quaternionnya. Ilustrasi SLERP terlihat pada dalam gambar 3. Bentuk umum dari *spherical interpolation* untuk vektor didefinisikan sebagai berikut [3] :

$$q_t = \frac{\sin(1-t)\theta}{\sin\theta} q_1 + \frac{\sin t\theta}{\sin\theta} q_2 \qquad \text{......................................(4)}$$



Gambar 3. Visualisasi SLERP

Nilai sudut θ dengan menghitung *dot product* antara quartenion q_1 dan q_2 sebagai berikut :

$$\cos\theta = \frac{q_1 \cdot q_2}{|q_1||q_2|}$$

$$= \frac{s_1 s_2 + x_1 x_2 + y_1 y_2 + z_1 z_2}{|q_1||q_2|}$$

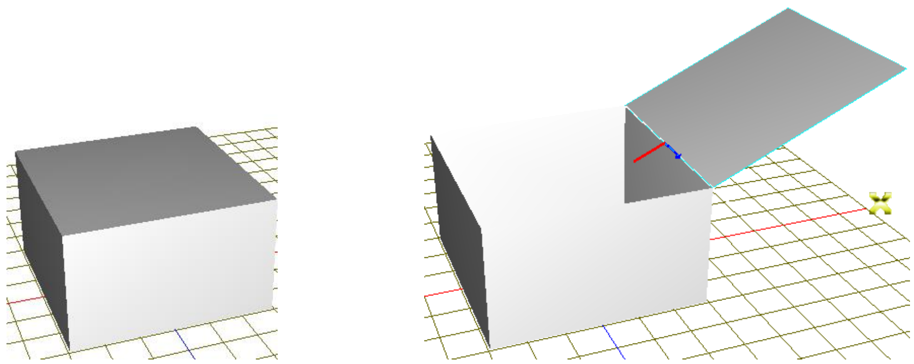
$$\theta = \cos^{-1}\left(\frac{s_1 s_2 + x_1 x_2 + y_1 y_2 + z_1 z_2}{|q_1||q_2|}\right)$$

Dimana sudut θ merupakan sudut yang dibentuk oleh kedua orientasi quartenion q_1 dan q_2. Fungsi SLERP dapat menghitung perpindahan sebuah vektor secara 3 dimensi sehingga implentasi ke bidang animasi menjadi berguna, seperti terlihat listing fungsi SLERP pada bahasa C. Hal ini dikarenakan dalam animasi, sebuah objek bergerak tidak hanya pada satu garis lurus, namun bergerak pada bidang tiga dimensi

```
quat slerp(quat qa, quat qb, double t)
{
//kalkulasi sudut antara qa dan qb
double cosHalfTheta = qa.w * qb.w + qa.x * qb.x + qa.y * qb.y + qa.z *
qb.z;
// Kalkulasi sudut
double halfTheta = acos(cosHalfTheta);
double sinHalfTheta = sqrt(1.0 -cosHalfTheta*cosHalfTheta);
double ratioA = sin((1 - t) * halfTheta) / sinHalfTheta;
double ratioB = sin(t * halfTheta) / sinHalfTheta;
//kalkulasi bilangan quaternion
qm.w = (qa.w * ratioA + qb.w * ratioB);
qm.x = (qa.x * ratioA + qb.x * ratioB);
qm.y = (qa.y * ratioA + qb.y * ratioB);
qm.z = (qa.z * ratioA + qb.z * ratioB);
return qm;
}
```

2.5 Eksperimen

Pada eksperimen penelitian ini, interpolasi obyek 3D menggunakan obyek primitif balok. Pada frame-0, terlihat bidang atap balok pada posisi menutup dengan nilai quartenion **q_1=1+0i+0j+0k**. Sedangkan pada frame-10 terlihat bidang atap balok pada posisi terbuka dengan nilai quartenion **q_2=0.21+0i+0j-0.98k**, seperti tampak pada gambar 4. Penelitian ini dilakukan untuk mendapatkan frame hasil interpolasi yang menghaluskan gerakan dari frame-0 sampai frame-10. Hasil quartenion interpolasi terdapat pada frame-1 sampai frame-9, seperti telihat pada tabel 1.

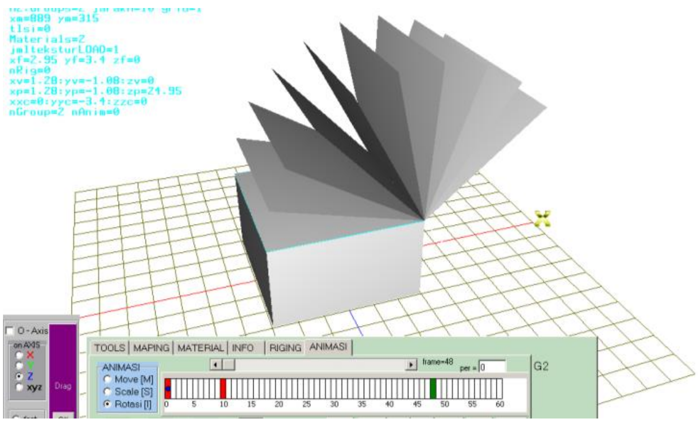


Frame-0 (**q_1=1+0i+0j+0k**) Frame-10 (**q_2=0.21+0i+0j-0.98k**)

Gambar 4. Dua frame obyek 3D akan ditentukan obyek-obyek interploasinya

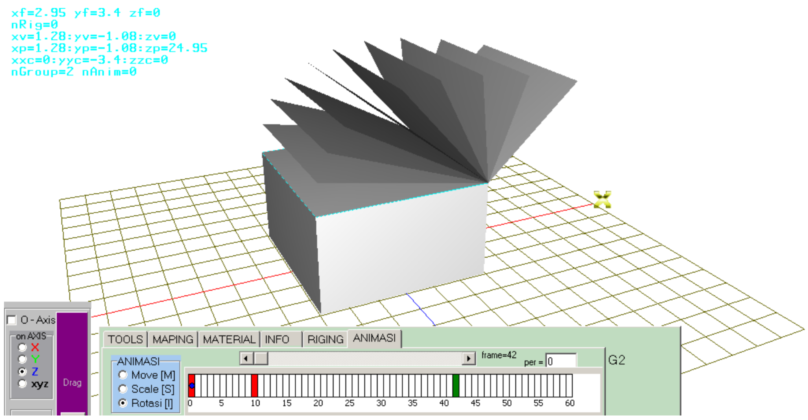
 ISSN 2085-4218
ITN Malang, 3 Pebruari 2018

Interpolasi proses animasi membuka atap balok menggunakan SLERP dihitung dari persamaan (4), hasilnya terlihat pada gambar 5. Disini terlihat obyek-obyek hasil interpolasinya berupa atap balok yang membuka secara perlahan.



Gambar 5. Hasil interpolasi menggunakan fungsi SLERP

Sedangkan hasil interpolasi proses animasi membuka atap balok menggunakan LERP yang dihitung dari persamaan (3) terlihat pada gambar 6.



Gambar 6. Hasil interpolasi menggunakan fungsi LERP

Frame-0 sebagai frame awal dan frame-10 sebagai frame akhir. Sedangkan frame-1 sampai frame-9 merupakan frame hasil interpolasi. Pada tabel 1 hasil eksperimen menunjukkan nilai vektor quartenion hasil interpolasi fungsi SLERP dan fungsi LERP, ditampilkan perubahan orientasi arah vektor i, j, dan k terjadi di semua orientasi arah. Terlihat hasil interpolasi fungsi SLERP lebih halus di semua orientasi arah, hal ini dapat dilihat dari hasil visualisasinya pada gambar 5.

Tabe1. Hasil pengujian fungsi SLERP dan fungsi LERP

Frame	Fungsi SLERP	Fungsi LERP
0	**1+0i+0j+0k**	**1+0i+0j+0k**
1	0.99+0i+0j-0.14k	0.92+0i+0j-0.1k
2	0.96+0i+0j-0.27k	0.84+0i+0j-0.2k
3	0.92+0i+0j-0.4k	0.76+0i+0j-0.29k
4	0.86+0i+0j-0.52k	0.68+0i+0j-0.39k
5	0.78+0i+0j-0.63k	0.61+0i+0j-0.49k
6	0.69+0i+0j-0.73k	0.53+0i+0j-0.59k
7	0.58+0i+0j-0.81k	0.45+0i+0j-0.68k

8	0.47+0i+0j-0.88k	0.37+0i+0j-0.78k
9	0.34+0i+0j-0.94k	0.29+0i+0j-0.88k
10	**0.21+0i+0j-0.98k**	**0.21+0i+0j-0.98k**

Terlihat bahwa fungsi SLERP atau *spherical linear interpolation* memiliki hasil interpolasi lebih bagus daripada fungsi LERP atau *linear interpolation* karena bentuk animasi tidak mengalami cacat obyek, terlihat pada gambar 5. Hasil interpolasi LERP memiliki cacat ukuran atap balok (menyusut) karena orientasi putar yang tidak konsisten ukurannya, seperti terlihat pada gambar 6.

3. Simpulan

Dalam bidang animasi fungsi SLERP atau spherical linear interpolation dalam *computer animation* sangatlah membantu perkembangan industri animasi sampai sepesat sekarang. Fungsi SLERP merupakan salah satu dari fungsi interpolasi pada computer animation yang lebih bagus daripada fungsi LERP. Perpindahan sebuah objek dihitung menggunakan aplikasi dari rotasi vektor menggunakan bilangan quaternion. Lebih jauh fungsi SLERP juga memiliki pengembangan interpolasi kurva Beizer yang dapat menghitung perpindahan lebih dari dua bilangan quaternion.

Daftar Pustaka

[1]. J. Vince, *Quartenions for Computer Graphics*. 1st ed. London: Springer, 2011.
[2]. V.Kremer, *Quaternions and SLERP*, Department for Computer Science, 2008.
[3]. David Eberly, *A Fast and Accurate Algorithm for Computing SLERP*, International Journal of Computer Mathematics Volume 86, 2009 - Issue 1.
[4]. Raheleh Ghadami, *Fast Methods for Spherical Linear Interpolation in Minkowski Space*, Advances in Applied Clifford Algebras,December 2015, Volume 25, Issue 4, pp 863–873.