A SYSTEM FOR ANATOMICAL REDUCTION OF BONE FRACTURES

Technical Field

The present application relates to a system for anatomical reduction of bone fractures.

Background to the Invention

For optimum healing of bone fractures in the human body, to ensure that the bone and surrounding joints are able to function correctly again, the fragments of the broken bone must be subjected to an anatomical reduction, which involves positioning and aligning the fragments of the broken bone to reconstruct the fractured bone as precisely as possible, so that the bone recovers to a form as close as possible to its original form as it heals.

This anatomical reduction may be performed by open surgery, in which large incisions are made in flesh around the affected joint and the bone fragments are manipulated by a surgeon to reposition and realign them as precisely as possible. Whilst this technique can be effective, it has disadvantages. For example, the anatomical reduction is not always perfect, in that the bone fragments are not always perfectly positioned or aligned. This leads to imperfect healing and can cause arthritis later in the patient’s life. Additionally, the extensive exposure required by the open surgery procedure typically slows bone healing and produces unsightly scars, as well as giving rise to an increased risk of infection. A prolonged period of post-operative rehabilitation is required, which requires the patient to endure an extended stay in hospital.

In order to mitigate the disadvantages of open surgery techniques, minimally invasive percutaneous procedures have been developed. These techniques involve sequentially
fixating and manipulating each bone fragment manually, without making large incisions in the patient’s flesh. Such techniques are associated with a faster recovery and a lower risk of infection compared to open surgery techniques. However, minimally invasive techniques may involve lower reduction accuracy, and in some cases the reduction accuracy is less than the minimum accuracy (typically <1mm translationally and <5 degrees rotationally) required for optimum clinical outcomes. Moreover, minimally invasive procedures require multiple radiographic images of the patient to be taken during the surgical procedure to ensure that the bone fragments are being correctly positioned and orientated during the procedure. This exposes the patient and medical staff to undesirably high levels of radiation. Notwithstanding these multiple radiographic images, fragment reduction remains sub-optimal.

Accordingly, a need exists for a system which improves the level of reduction accuracy of minimally invasive surgical techniques and reduces exposure to radiation, whilst retaining the recovery speed and low infection risk of existing minimally invasive techniques.

**Summary of Invention**

The present application relates to a system for anatomical reduction of bone fractures in which first and second manipulators, and optionally a third manipulator, are attached to fragments of the fracture to be reduced by percutaneous attachment devices such as Schanz pins. An underlying processing system determines, from one or more medical images of the fracture, manipulations such as rotations and translations of the bone fragments required to correctly reposition and align the fragments for optimum healing of the fracture. The processing system provides motion reference signals (position, speed, acceleration) for a controller, which in turn causes the first, second and third manipulators to effect the calculated manipulations.
According to a first aspect of the present invention there is provided a system for reduction of bone fractures, the system comprising: a first manipulator for manipulating a bone section of the fracture, the first manipulator being attachable to the bone section by means of a percutaneous attachment device; a second manipulator for manipulating a first bone fragment of the fracture, the second manipulator being attachable to the first fragment by means of a percutaneous attachment device; and a processing system configured to determine reference signals for the first and second manipulators required to effect manipulations of the bone section and the first fragment required for correct anatomical reduction of the fracture; and a controller configured to control the first and second manipulators to cause them to perform the manipulations of the bone section and the first bone fragment.

The system of the first aspect of the present invention permits minimally invasive semi-automated anatomical reduction of intra-articular joint and other fractures such as long bone fractures. By determining manipulations required for correct reduction of the fragment using processing means, and implementing the manipulations using manipulators controlled by control means greater reduction accuracy can be achieved than with conventional manually performed minimally invasive surgical techniques.

Preferably, the first manipulator is a spatial parallel manipulator comprising first and second end sections connected by a plurality of linear actuators.

The system may further comprise a third manipulator for manipulating a second bone fragment of the fracture, the third manipulator being attachable to the fragment by means of a percutaneous attachment device, wherein the processing system is configured to determine reference signals for the third manipulator required to effect manipulations of the second fragment required for correct anatomical reduction of the fracture and the controller is further configured to control the third manipulator to perform the manipulations of the second bone fragment.
Preferably, the second and/or third manipulator is a spatial parallel manipulator comprising a fixed base and a moveable platform connected to the fixed base by a plurality of linear actuators.

For example, the fixed base may be connected to the moveable platform by six linear actuators.

The controller may be configured to cause the manipulators to perform the manipulations of the reference bone and the bone fragment(s) substantially simultaneously.

The processing system may be configured to: receive an image of the fracture; segment the image of the fracture to identify fracture surfaces of the reference bone and the bone fragment(s) of the fracture; generate a fragment surface layer for the reference bone and the fragment(s) representative of surfaces of the reference bone and the bone fragments; display a graphical representation of the reference bone and the bone fragments; receive a user input; manipulate the graphical representation to simulate reduction of the fracture based on the user input received; record the manipulations of the graphical representation; and, based on the recorded manipulations, determine the reference signals for the manipulations required for correct anatomical reduction of the fractured bone.

The user input may be received by means of a virtual joystick presented as part of a graphical user interface by the processing system.

According to a second aspect of the invention there is provided a method for calculating manipulations required to effect an anatomical reduction of a bone fracture, the method comprising the steps of: segmenting an image of the fracture to identify fracture surfaces of bone fragments of the fracture; generating a fragment surface layer for each fragment representative of surfaces of the bone fragment; calculating an axis of each bone fragment; calculating fracture surfaces for each bone fragment; and calculating, based on
the axes and fracture surfaces calculated, manipulations required for reduction of the fracture.

The segmenting of the image of the fracture may comprise calculating Hounsfield intensity values for structures shown in the image and identifying the fracture surfaces from the Hounsfield intensity values.

Generating the fragment surface layer for each fragment may comprise using data from the segmented image to generate a point cloud representing each bone fragment and performing a triangulation on the point cloud.

Calculating the axis of a bone fragment may comprise defining two sections of the bone fragment, calculating the point centre of each of the defined sections of the fragment and calculating an axis vector for the bone fragment by subtracting the point centre of one section from the point centre of the other section.

Calculating the point centre of a section of the bone fragment may comprise calculating the mean x, y and z coordinates for each point of the point cloud in the section.

Calculating the manipulations required for reduction of the fracture may comprise calculating a manipulation required for axial alignment of the fragments, calculating an angle of rotation required for fracture surface alignment and calculating a translation required to close a gap between the bone fragments.

Calculating the manipulation required for axial alignment of the fragments may comprise calculating a transformation required to align the axis vectors of the fragments.

Calculating the angle of rotation required for fracture surface alignment may comprise generating a first polyline representative of the fracture surface of a first, reference fragment; generating a plurality of second polylines representative of the fracture surface
of a second fragment, each second polyline having an incremental angular offset with respect to the first polyline; performing a comparison of the first and each of the plurality of second polylines, wherein the angle of rotation required is determined by calculating the angular offset for which the comparison determines that the first and second polylines are most similar.

Performing the comparison may comprise: calculating a first cross correlation coefficient for the first polyline; for each of the plurality of second polylines, calculating a second cross correlation coefficient; and calculating a difference value between the first and second cross correlation coefficients, wherein the angle of rotation required is the angular offset of the second polyline for which the difference value is smallest.

According to a third aspect of the invention there is provided a method for calculating manipulations required to effect an anatomical reduction of a bone fracture, the method comprising the steps of: segmenting an image of the fracture to identify fracture surfaces of bone fragments of the fracture; generating a fragment surface layer for each fragment representative of surfaces of the bone fragment; displaying a graphical representation of the bone fragments; receiving a user input; manipulating the graphical representation of one of the bone fragments to simulate reduction of the fracture based on the user input received; recording the manipulations of the graphical representation of the bone fragment; and, based on the recorded manipulations, determining reference signals for manipulations of the fragments of fractured bone required for correct anatomical reduction of the fractured bone.

The user input may be received by means of a virtual joystick presented as part of a graphical user interface.

The processing means of the system of the first aspect may be configured to perform the method of the second or third aspects, and the controller may be configured to cause the manipulators to perform the calculated manipulations.
According to a fourth aspect of the invention there is provided a computer program which, when executed on appropriate processing means, performs the method of the second or third aspect.

**Brief Description of the Drawings**

Embodiments of the invention will now be described, strictly by way of example only, with reference to the accompanying drawings, of which:

Figure 1 is a schematic perspective representation of a part of a system for anatomical reduction of bone fractures;

Figure 2 is an alternative view of the system illustrated in Figure 1;

Figure 3 is a schematic representation of a first manipulator used in the system illustrated in Figures 1 and 2;

Figure 4 is a schematic representation of a second and third manipulator used in the system illustrated in Figures 1 and 2;

Figure 5 is a flow diagram illustrating steps taken prior to and during a surgical procedure to reduce a part distal femoral fracture performed using the system of Figures 1 to 4;

Figure 6 is a schematic representation of a distal femoral bone fracture;

Figure 7 is a CT image showing a horizontal view of a bone fracture that has been processed to show Hounsfield intensities of different portions of the fracture;

Figure 8 is a point cloud representing a two-part long bone fracture (diaphyseal fracture);
Figures 9 and 10 illustrate steps taken in determining an axis of a bone fragment as part of a process for reducing the fracture;

Figure 11 illustrates a graphical representation of a two part diaphyseal fracture for which the axes of the bone fragments have been aligned;

Figure 12 illustrates the result of a step of angular alignment of the bone fragments illustrated in Figures 8 to 11;

Figure 13 is a schematic representation of the second manipulator illustrated in Figure 4;

Figure 14 is a flow chart illustrating steps performed as part of a process to determine a reference signals of linear actuators of the manipulators illustrated in Figures 1 and 2 to achieve a desired and defined manipulation of a bone fragment; and

Figure 15 is a flow chart illustrating steps performed to calculate Forward Kinematics of the manipulators illustrated in Figures 1 and 2.

Figures 16 to 18 are screenshots of a graphical user interface used in an alternative embodiment of a system for anatomical reduction of fractures.

Description of the Embodiments

Figures 1 to 3 are schematic illustrations of part of a system for anatomical reduction of bone fractures. The system, illustrated generally at 10, includes a first manipulator 12 for restoring limb length and alignment and for decompressing the fracture, a second manipulator 14 for manipulating a first fragment of a fractured bone and a third manipulator 16 for manipulating a second fragment of fractured bone. In the case of two-part fractures the third manipulator 16 may not be required.
The system 10 also employs a motion recording system which uses up to eight cameras (not shown) to register motion of bone fragments. The motion recording system uses a reference marker frame 20 carrying retro-reflective markers that is rigidly attached to a selected reference bone of the patient, and an active marker frame 18 carrying retro-reflective markers rigidly attached to a bone fragment that is to be aligned as part of the reduction of the fracture. Additional active marker frames 21, 23 carrying retro-reflective markers are rigidly attached to the first and second bone fragments that are to be manipulated (e.g. rotated and translated) as part of the reduction of the fracture. The reference marker frame 20 and the active marker frames 18, 21, 23 can be detected by the camera(s) of the motion recording system, with the reference marker frame 20 providing a reference or origin against which movement of the active marker frames 18, 21, 23 (and therefore movement of the bone fragments to be reduced) can be recorded. The motion recording system is able to record movement of the active marker frames 18, 21, 23, and therefore of the bone fragments to which the active marker frames 18, 21, 23 are attached, in six degrees of freedom.

Figures 2 and 3 are alternative views of the system 10 illustrated in Figure 1, from which it can be more clearly seen that the first manipulator 12 in this example is a lightweight spatial parallel manipulator in the form of a Gough Stewart platform made up of first and second parallel, partially annular, end sections 22, 24 connected by a number (in this example 6) of linear actuators 26. The first end section 22 of the first manipulator 12 is fixed such that changes in the length of the linear actuators 26 cause the second end section 24 to move relative to the first end section 22. As is explained below, the first manipulator 12 remains in place during a surgical procedure for reducing a fracture. The procedure requires high resolution medical images such as CT images to be taken, and thus the first manipulator should be constructed of a material that does not impede the taking of such images. The first manipulator 12, which is also referred to as an “external robot”, is configured to extend and align long bones around the fracture that is to be reduced. To this end, the
first manipulator 12 can be connected to the long bones around the fracture site by means of pins and wires (not shown), each having a first end which is screwed into or inserted in a hole drilled in the long bone by a surgeon, and a second end that is attached to the first end section 22. The length of the linear actuators 26 of the first manipulator 12 can then be adjusted to achieve the desired spacing and angulation between the first and second end sections 22, 24 of the of the first manipulator 12, which in turn extends and aligns the long bones around the fracture. In the example illustrated in Figure 2 the first manipulator 12 is used to align the tibia with respect to the femur.

By extending and aligning long bones using the first manipulator 12 in this way, a workspace is created around the fracture in which the second manipulator 14 (and the third manipulator 16, if appropriate) can operate to rotate and translate a bone fragment, as is described in more detail below. As the first manipulator 12 remains attached to the patient’s bones following the surgical procedure, it assists in maintaining the correct position and orientation of bone fragments during post-operative healing of the fracture.

Figure 4 is a schematic illustration showing the second manipulator 14 in more detail. The third manipulator 16 is similar in construction and operation to the second manipulator 14. The second manipulator 14, which is also referred to as an “internal robot”, a lightweight spatial parallel manipulator, and is configured to manipulate a fragment of bone in and around the fracture site by effecting rotational and translational movement of the bone fragment so as to achieve the correct position and orientation for a successful reduction of the fracture, that is to say reconstruction of the fractured bone with the fragments in the correct position and orientation.

The second manipulator 14 and the third manipulator 16 each take the form of a hexapod robot, having a platform 30 on which an end effector 32 is mounted. The platform 30 is connected to a fixed partially annular base 34 by means of six linear actuators 36 (hence the term “hexapod robot”). Each linear actuator 36 is connected at one end to the base 34 by means of a f=2 universal joint 38 and at the other end to the platform 30 by means of...
a f=2+1 universal joint 40. This arrangement permits precise movement of the end effector 32 with up to six degrees of freedom. The end effector 32 itself is mounted for rotation about its longitudinal axis, whilst the base 34 is attached to a linearly moveable table 42 to increase the workspace of the second manipulator 14 and to permit linear movement of the second manipulator 14 in a plane generally parallel to the base 34.

The linear actuators 36, the end effector 32 and the linearly moveable table 42 are electrically operable components which are connected to a controller (not shown) which controls the movements of the second manipulator 14 to achieve highly precise and accurate rotation and translation of a bone fragment to which the end effector 32 is attached by means of a pin such as a Schanz pin inserted into the bone fragment.

Similarly, the linear actuators 26 of the first manipulator 12 are also electrically operable, and are connected to the controller, so as to permit precise control of the degree of extension of each strut 26, and therefore the spacing and angular displacement between the first and second end sections 22, 24 of the first manipulator 12 to achieve a desired alignment and spacing of the long bones around the fracture site.

The controller receives signals from an underlying processor or processing system such as a general purpose computer running appropriate software (not shown) which calculates the rotations and translations required to position and align (i.e. reduce) the bone fragment manipulated by the second manipulator 14 (and the bone fragment manipulated by the third manipulator 16, where provided) correctly for optimum healing, based on images, such as CT (computer tomography) scans, of the fracture and the surrounding tissue taken before and during the reduction procedure. The processing system also calculates reference signals required by the manipulators 14, 16 to effect the manipulations calculated. Each manipulator 14, 16 has a feed drive controller which is configured to minimise the error between the reference input signals received from the controller of the processing system and positional feedback signals received from an absolute displacement transducer associated with each actuator 36.
The processing system also calculates the alignment and extension of the long bones around the fracture site required to restore limb length and limb alignment for optimum healing of the fracture, again based on medical images taken of the fracture and the surrounding tissue before and during the reduction procedure, and Cartesian motion feedback from the motion recording system. The processing system also calculates and transmits reference signals indicative of the required alignment and extension of the long bones to a position controller of the first manipulator 12 to cause it to adjust the length of one or more of the linear actuators 26 so as to effect the calculated alignment and extension of the long bones around the fracture site, as will be described in more detail below. The first, second and third manipulators 12, 14, 16 may be controlled simultaneously to effect their respective manipulations on the fracture in parallel.

Figure 5 is a flow diagram illustrating steps taken prior to and during a surgical procedure performed using the system of Figures 1 to 4. In this example the system is used to perform a minimally invasive surgical reduction of a three part intra-articular distal femoral fracture of the type illustrated in Figure 6.

In a first step 100 a CT scan is taken of a suspected fracture and surrounding tissue for the purpose of diagnosing the patient’s injury. This CT scan enables a surgeon to identify fragments (shown as F1 and F2 in Figure 6) of a fractured bone. Additionally, data from this pre-operative CT scan enables a medical engineer to digitally segment the fracture in a preliminary step, to assist in the identification of bone fragments.

In the operating theatre, Schanz pins are inserted in the identified bone fragments by the surgeon (step 102), by means of which the bone fragments can be manipulated by the second manipulator 14 and, if appropriate, the third manipulator 16. In step 104 a reference bone (in this example the femur, shown as F3 in Figure 6) is fixated (i.e. attached) to the first manipulator 12 by means of Schanz pins inserted into the reference bone and attached to the first end section 22 of the first manipulator 12. The reference
bone F3, to which the marker frame 20 is rigidly attached by means of a pin, will be used as a reference point for the assembly of the two other fragments F1 and F2. A further, inter-operative, CT scan of the fracture site is then taken to ensure that the locations and dimensions of the fragments F1 and F2 have not changed with respect to the previous scan.

At step 106, 3D image data from a further CT scan and from the motion recording system are registered by the processing system. This step is required to ensure that both data sets are referenced to the same coordinate frames, to ensure that rotations and translations of bone fragments performed during the procedure have the expected effect.

In step 108 features such as surfaces, points, contours and the like of the bone fragments present in the image are identified and extracted for use in a later matching step in which pairs of bone fragment which match, i.e. belong together, are identified and the appropriate rotations and translations of the matching bone fragment pairs to achieve the optimum reduction of the fracture are determined. The matching step is not required where there are only two bone fragments, as the two fragments will clearly belong to a single pair, but for multiple fragment fracture cases matching is required.

The features may be identified and extracted automatically, or may be identified by a qualified orthopaedic surgeon using a user interface which presents the images of the bone fragments and permits the surgeon to manipulate the images of the bone fragments, or to select matching features of the bone fragments, or both, for use in determining the required rotations and translations.

At step 110 the identified features of the bone fragments are used by the processing system in an automatic registration process which calculates the required translations and rotations of the bone fragments to achieve the desired optimum reduction of the fracture. This automatic registration process is described in more detail below.
Once the required translations and rotations of the bone fragments have been calculated, they are passed, at step 112, to the controller. The controller causes the first manipulator 12 to create a workspace in which the second manipulator 14 (and if appropriate the third manipulator 16) are able to work, by decompressing the fracture by lateral extension of the reference bone. The processing system then translates the translations and rotations into control signals that are used by the controller to control the second manipulator 14 (and, if appropriate, the third manipulator 16) to cause the end effector 32 to effect the required translations and rotations of the bone fragment to which it is attached. The controller also causes the first manipulator 12 to restore the active bone (i.e. the bone to which active marker frame 18 is attached) to its correct position and orientation with respect to the reference bone F3.

Once the translations and rotations have been effected by the second manipulator 14 and the active bone (in this example the tibia) has been restored to its correct position and orientation by the first manipulator 12 a further CT scan is taken, at step 114, to verify that the reduction of the fracture has been achieved to the desired level of accuracy. If the reduction has not been achieved to a satisfactory level of accuracy steps 106 to 114 are repeated. If the reduction has been achieved to a satisfactory level of accuracy the bone fragments are fixated manually by the surgeon with respect to the first manipulator 12 at step 118 and a further CT scan is taken at step 120, to ensure that the fragments have not been displaced during or after the manual fixation step.

The process used by the system 10 for reduction of a bone fracture, used in steps 106 to 110 of the surgical procedure illustrated in Figure 5 will be now described in more detail with reference to Figures 7 to 12.

As is discussed above, a first part of the reduction process involves the segmentation of bone from soft tissue in the image produced by the pre-operative CT scan. This segmentation is carried out by the processing system, which calculates Hounsfield intensities of the structures shown in the images produced by the CT scan. As is well
known, different structures in the human body have different Hounsfield intensities when imaged by a CT scan. For example, the inner part of bone, shown at 150 in Figure 7, has a lower density than the surface part of bone 152. This property can be used by the processing system to identify and segment fracture surfaces from other sections of the bone, since fracture surfaces have Hounsfield intensities similar to those of the inner section of the bone.

Once the bone fragments have been identified by the segmentation step, a matching step may be required to match a fracture surface of one bone fragment with a corresponding fracture surface of another bone fragment. This matching step may not be required for fractures where there are only two fragments, such as simple long bone fractures, since the fracture surfaces of the two fragments must match. However, for more complex fractures having more than two fragments matching using extracted bone features is typically required.

This matching step may be performed by a qualified orthopaedic surgeon using a user interface which presents the images of the bone fragments and permits the surgeon to identify the matching fracture surfaces of different bone fragments based on features such as contours, points or extracted and calculated surface areas in mm². Alternatively, the processing system may automatically identify the features automatically, and may use these features to identify matched pairs of fracture surfaces.

Once the CT images have been processed to identify and segment the bone fragments, a point cloud of the segmented 3D bone shape is generated, based upon data from the segmented images of the fracture, for each bone fragment. Each point cloud is a matrix of Cartesian coordinates of points representing the surface of the bone fragments, and can be represented graphically, as is shown at 170 in Figure 8. It will be noted that the point cloud illustrated in Figure 8 represents a long bone fracture (in this instance a diaphyseal fracture), but it will be appreciated that the techniques described herein are equally
applicable to other fracture cases, for example restoration of bone length and alignment of the two part femoral fracture illustrated in Figure 6.

Each point of the point cloud has an $x$ coordinate, a $y$ coordinate and a $z$ coordinate, and these coordinates are stored in a matrix having 3 columns ($x$, $y$, $z$ coordinates) and $n$ rows, where $n$ is the number of points in the point cloud. The processing system performs a triangulation on this point cloud to generate a fragment surface layer, as shown in Figure 11, which is representative of the surfaces of the bone fragments.

The processing system calculates the axis and fracture surfaces of the bone fragments identified in the point cloud. In a first step each bone fragment 172, 174 is defined in two sections (referred to as section A and section B), and the point centre $c_i$ (not shown) of each of the sections A and B of each bone fragment is calculated.

The processing system calculates the point centre $c_i$ of each section A and B of each bone fragment by calculating the mean position of the $x$, $y$ and $z$ coordinates of the $n$ points in the point cloud for the selected section. The position of the point centre $c_i$ of a section of a bone fragment is at the mean $x$, $y$ and $z$ coordinates of the selected section of the bone fragment.

Vector algebra is then used by the processing system to calculate an axis vector for each bone fragment 172, 174 by subtracting the coordinates of the point centre of one section of the bone fragment 172, 174 from the point centre of the other section of the bone fragment 172, 174, as indicated in the following Matlab code:

```
axis_vA=c2-c1 %axis vector 1st fragment
axis_vB=c2-c1 %axis vector 2nd fragment
```

These axis vectors (shown in Figure 10) are used in a later step to align the bone fragments specifying four degrees of freedom, namely along two Cartesian axes X and Y.
and around two Cartesian axes $\Theta_x$ and $\Theta_y$. This permits lateral and rotational alignment of the bone fragments 172, 174.

The shaft axis of each bone fragment 172, 174 is used by the processing system to extract (i.e. calculate) and plot fracture surfaces of the bone fragment 172, 174. This extraction of the fracture surface is performed based on the assumption that all undamaged surface unit vectors are perpendicular to a unit vector of the calculated axes of the two bone fragments 172, 174. An example Matlab implementation of an algorithm used by the processing system to extract and plot the fracture surface is presented below:

```matlab
v1=tnorm1A; %array containing unit vectors of each triangulated surface element (1st fragment)
v2=axis_v_nA; %unit vector of axis vector 1st fragment

%calculate angles between unit axis vector and each surface unit vector
for i=1:length(v1)
    angle(i) = atan2(norm(cross(v1(i,:),v2)),dot(v1(i,:),v2));
    angle(i)=angle(i)*180/pi;
end

%apply constraints to extract fracture surface points for fragment top and bottom section
for i=1:length(tnorm1A)
    if angle(i)<=70 && angle(i)>=100
        plot3(P0A(i,1),P0A(i,2),P0A(i,3),'x');
        if P0A(i,3)>110
            ext_stA(i,:)=P0A(i,1:end);
        elseif P0A(i,3)<80
            ext_sbA(i,:)=P0A(i,1:end);
        end
    end
end
```

The algorithm starts by calculating the angle between the unit axis vector and each surface unit vector. Having calculated this angle it can be used as an evaluation criteria to extract surface points for the fragment top (section A) and bottom section (section B) by using two if-functions and by specifying numerical values in order to separate the fragment top and bottom section. The extraction of the surface points is performed under the assumption that all undamaged surface unit vectors are perpendicular to a unit axis vector. If this is not true, surface points must also be points within the fractured surface region.

Having extracted physical quantities such as the axis vector and the fracture surface points of the two fragments the processing system can calculate the rotations and translations required to reduce (i.e. reconstruct) the fractured bone. Typically, such as in the example of a long bone tibia fracture illustrated in Figures 7 to 12, this is performed in three steps, namely the registration or alignment of the shaft axes of the fragments, registration of the shaft axis rotation (i.e. a rotation about the axis of a fragment to required for correct alignment of the fracture surfaces) and registration of the distance between the fragments (i.e. a translation required to minimise the distance between the fracture surfaces).

In order to register (i.e. reduce or realign) bone fragments based on their shaft axes a coordinate frame is generated and associated with each fragment 172, 174 by the processing system. The coordinate frame for the first fragment 172 is defined by first selecting the calculated axis vector of the fragment 172 as one of the axes of the coordinate frame. In this example, the axis calculated for the fragment 172 is selected as the X axis of the coordinate frame for that fragment. The same steps are performed for the second fragment 174.

The cross-product of the defined X axes of the fragments 172, 174 gives the Z axis of the coordinate frame, and the cross product of the X axis of the fragment 172, 174 for which
the coordinate frame is defined and the $Z$ axis determines the $Y$ axis of the coordinate frame.

Exemplary Matlab code for defining a coordinate frame for fragments of a diaphyseal fracture is presented below:

\[
\begin{align*}
A &= \text{axis}_v_nA; \quad \text{unit axis vector Fragment A (X-axis)} \\
B &= \text{axis}_v_nB; \quad \text{unit axis vector Fragment B (X-axis)} \\
Z &= \text{cross}(A, B); \quad \text{calculate cross product of A and B to obtain Z axis vector of frame} \\
LL &= \text{norm}(Z); \quad \text{Calculate magnitude of Z axis vector} \\
Z_n &= Z/LL; \quad \text{Calculate unit axes vector of axis vector Z} \\
\text{Cy} &= \text{cross}(Z_n, B); \quad \text{calculate cross product of unit axis vector Z and unit axis vector B to obtain Y axis} \\
LLL &= \text{norm}(\text{Cy}); \quad \text{Calculate magnitude of Y axis vector} \\
Y_n &= \text{Cy}/LLL; \quad \text{Calculate unit axes vector of axis vector Y} \\
FB &= \begin{bmatrix} B & Y_n' & Z_n' \end{bmatrix}; \quad \text{assembly 3x3 Coordinate Frame Matrix using unit axes vectors of X, Y and Z axes}
\end{align*}
\]

The coordinate frame for each bone fragment 172, 174 specifies the orientation of the fragment 172, 174 relative to a global body coordinate system (BCS) defined by the matrix

\[
BCS = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}
\]
All of the surface points (from the point cloud) of each bone fragment 172, 174 are referenced with respect to the BCS frame matrix and therefore represent position vectors with respect to the BCS frame matrix ($^{BCS}p_B_i$ is the position vector for the fragment 174, $^{BCS}p_A_i$ is the position vector for the fragment 172). This allows the position vectors $^{BCS}p_B_i$ and $^{BCS}p_A_i$ of the surface points of each fragment 172, 174 to be rotated so that they are defined with respect to the coordinate frame matrix FA or FB of that fragment 172, 174 rather than the global BCS frame matrix. This relationship is shown in the formula below applying the inverse of the defined coordinate frame matrix to rotate the vectors $^{BCS}p_B_i$ and $^{BCS}p_A_i$.

$$^{FB}p_B_i = \left[^{FB}FB\right]^{-1} \cdot ^{BCS}p_B_i$$

The resulting axial alignment is plotted, as shown in Figure 11. Exemplary Matlab code to implement this formula is shown below:

```matlab
%% Rotate in Fragment A space
for i=1:length(pA_FA)
    pA_FAr=inv(RRA)*pA_FA(i,:)';
    pA_FAr2(i,:)=pA_FAr';
end
plot3(pA_FAr2(:,1),pA_FAr2(:,2),pA_FAr2(:,3),'ro'), hold on
```

Once the matrix transformation required to achieve the desired axial alignment of the bone fragments 172, 174 has been calculated, the processing system calculates the rotational alignment of the bone fragments 172, 174 around the Cartesian X-axis.

To do this the processing system calculates a reference two-dimensional polyline approximating a two dimensional fracture surface plotted perpendicular to the calculated axis vector X of a selected one of the bone fragments 172, 174, which is used as a reference. In the present example the first fragment 172 is used as the reference...
fragment. A plurality of further two-dimensional polylines are calculated, approximating a two dimensional surface of the other fragment (in this example the second fragment 174). Each of the plurality of further polylines has a rotational offset, which increases for each successive polyline in small increments between 0 and 360 degrees. By doing this, the optimum rotational axis alignment between the fragments 172 and 174 can be determined by comparing cross correlation coefficients $C_A$ of the reference polyline of the first fragment 172 with a cross correlation coefficient $C_{Bi}$ for the second fragment 174 for each rotational increment of the polyline.

The processing system then selects the angle of rotation for which the correlation coefficient $C_{Bi}$ is closest to the cross correlation coefficient $C_A$ (i.e. for which the difference value is smallest) as the correct angle of rotation for angular alignment of the bone fragment 174, 172 with respect to the reference bone fragment 172. Figure 12 illustrates correctly angularly aligned bone fragments 172, 174.

Once the angular rotation required for correct angular alignment of the bone fragments 172, 174 has been calculated, the processing system must calculate the translational movement between the fragments 172, 174 required for correct reduction of the fracture. The translational distance between the fragments 172, 174 can be minimised using the Iterative Closest Point algorithm developed by Besl and McKay (1992).

The translations and rotations of a bone fragment calculated by the processing system for correct reduction of a fracture must be facilitated by the second manipulator 14 and if appropriate the third manipulator 16. The translations and rotations are calculated in terms of a Cartesian coordinate system (operational space or Cartesian space), whereas the manipulators 14, 16 operate on a joint coordinate system (joint space) defined by the coordinates of the joints of the manipulators 14, 16. Therefore, a transformation between the operational space and the manipulator joint space is required, to enable the manipulators 14, 16 to effect the translations and rotations in the operational space calculated by the processing system, and to enable the processing system correctly to
determine the joint coordinates of the manipulators 14, 16 based Cartesian inputs (X, Y, Z, \( \theta_x, \theta_y, \theta_z \)) received from the processing system.

The transformation between Cartesian coordinates and the joint coordinates is referred to as Inverse Kinematics, whilst the reverse transformation, between joint coordinates and Cartesian coordinates, is referred to as Forward Kinematics. An iterative algorithm to solve the Forward Kinematics is described in Figure 15, whilst a process for the calculation of the Inverse Kinematics required to implement the translations and rotations required for reduction of a fracture is described below.

The entire structure of the second manipulator 14 is fully specified by four design parameters, namely the joint circle diameter \( d_p \) of the moveable platform 30, the joint circle diameter \( d_b \) of the fixed base 34, the angular joint spacing of the base \( \Theta_b \) and the platform \( \Theta_p \), assuming a standard 60 degree offset angle between the base and the platform.

The coordinate frame \{P\} of the platform 30 relative to the base 34, defined by a coordinate frame \{B\}, is defined by a position vector linking the origin of \{B\} and \{P\}.

Having defined the kinematic model of the manipulators by the four design parameters mentioned above, in a further pre-processing step the processing system calculates angular joint positions of a platform joint \( P_i \) (i.e. the joint linking the \( i \)th linear actuator 36 to the platform 30) and of a base joint \( B_i \) (i.e. the joint linking the \( i \)th linear actuator 36 to the base 36) as a function of the angular joint spacing \( \Theta_b \) and \( \Theta_p \). From the angular joint positions \( P_i, B_i \), joint vectors \( p_i \) and \( b_i \) relative to the origin of \{P\} and \{B\} can be calculated. A rotation matrix \( {}^B_P R \) is calculated based on Cartesian angular inputs \( \Theta_x, \Theta_y, \Theta_z \).
From these initial calculations the processing system can calculate the magnitude of vectors \( l_i \) (i.e. the magnitude of the vector \( l \) for the \( i \)th linear actuator 36 of the manipulator), which corresponds to the length of the \( i \)th linear actuator 36 using the loop closure equation:

\[
|l_i| = |^{\mathbb{B}}r + ^{\mathbb{B}}R \cdot ^{\mathbb{B}}p_i - ^{\mathbb{B}}p_i|,
\]

where \( ^{\mathbb{B}}p_i \) and \( ^{\mathbb{B}}b \) are vectors describing the geometry of the mechanism. The vector \( ^{\mathbb{B}}r \) is a position vector linking origin \{B\} and \{P\} and thereby specifying the translational movement of the platform with respect to the fixed base.

Having calculated the length of each of the linear actuators 36 for the position input the processing system calculates the joint reference parameters (position, speed and acceleration) for every sampling interval \( q, q, q, q, q, q \) and each linear actuator 36. This is shown in the flow diagram of Figure 14. These reference data are the transferred to the controller to drive each actuator 36 of the manipulator 14 and to minimize the error between reference data and position feedback received from an absolute displacement sensor associated with each actuator 36 of the manipulator 14.

In a first step 180, maximum velocity \( \dot{\Theta}_{\text{max}} \) and acceleration \( \ddot{\Theta}_{\text{max}} \) for each of the linear actuators 36 are defined by the operator of the system 10.

At step 182 the processing system solves the loop closure equation (Inverse Kinematics equation)

\[
|l_i| = |^{\mathbb{B}}r + ^{\mathbb{B}}R \cdot ^{\mathbb{B}}p_i - ^{\mathbb{B}}p_i|,
\]

to obtain the relative position changes \( s_i \) for the \( i \)th actuator (where \( i=1\ldots6 \)) between the initial magnitude of the leg vector \( |l_{i,j}| \) and the new magnitude of the leg vector \( |l_{i+1,j}| \) (representing the new length of the actuator 36).

In a further step 184 the processing system calculates parameters \( t_c, t_a, t_d \) of a trapezoidal velocity profile for the change in the length of each linear actuator 36. The change of length occurs in three phases: an acceleration phase, which takes place during time a period from a motion start time \( t \) to a time \( t_a \), a deceleration phase, which takes place
from a time $t_d$ to a motion end time $t_e$, and a constant velocity phase, which takes place during a time period from $t_a$ to $t_d$.

At step 186 the processing system determines the actuator which will have the longest operational travel time $t_{emax}$ to perform the positional change based on the parameters specified in steps 180 and 182. The actuator with the longest operational travel time becomes the leading robot axis. At step 188 the processing system compares this maximum travel time $t_{emax}$ to the calculated travel time end time $t_{e,i}$ for the $i$th linear actuator 36. If $t_{emax}$ is not equal to $t_e$ the joint velocities of the remaining linear actuators 36 of the manipulator are adjusted, in step 190 so that all of the actuators 36 start and end their motion at the same time, using the equation

$$
\dot{\Theta}_{i,adj} = \frac{\Theta_{max} t_e}{2} - \sqrt{\frac{\Theta_{max}^2 t_e^2}{4} - \left(s_{i,e} \dot{\Theta}_{max}\right)}
$$

Once this adjustment has been made, or if $t_{emax}$ is equal to $t_e$, processing passes to step 192, in which path data $s(t)$, $\dot{s}(t)$ and $\ddot{s}(t)$ (i.e. displacement as a function of time, linear velocity as a function of time and linear acceleration as a function of time) for the acceleration phase, the constant velocity phase and the deceleration phase are calculated using the following equations:

For the acceleration phase

$$
\dot{s}(t) = \ddot{\Theta}_{max}
$$

$$
\dot{s}(t) = \ddot{\Theta}_{max} t
$$

$$
s(t) = \frac{1}{2} \ddot{\Theta}_{max} t^2
$$

For the phase of constant velocity

$$
\ddot{s}(t) = 0
$$
\[
\ddot{s}(t) = \dot{\Theta}_{\text{adj},j}
\]
\[
s(t) = \left(\dot{\Theta}_{\text{adj},i} \right) \left(\frac{1}{2} \frac{\dot{s}^2}{\dot{\Theta}_{\text{max}}} \right)
\]

For the deceleration phase
\[
\ddot{s}(t) = -\ddot{\Theta}_{\text{max}}
\]
\[
\ddot{s}(t) = \dot{\Theta}_{\text{adj},j} - \ddot{\Theta}_{\text{max}}(t - t_d)
\]
\[
s(t) = \dot{\Theta}_{\text{adj},j}(t_{e\text{max}} - t_d) - \left(\frac{\ddot{\Theta}_{\text{max}}}{2(t_{e\text{max}} - t_d)^2}\right)
\]

Having calculated the path data, processing moves to step 194, in which the processing system calculates joint parameters \(q(t), \dot{q}(t)\) and \(\ddot{q}(t)\) using the path data and the sign function \(y = \text{sign}(x) = \{-1 \text{ for } x < 0, 0 \text{ for } x = 0, 1 \text{ for } x > 1\}\) to verify the joint parameters and to distinguish between positive and negative motions. With the parameters (extension \(s\), maximum speed and maximum acceleration) specified for the linear actuators 36 and the joint parameters calculated the processing system is able to cause the required synchronous movement of the linear actuators 36 to effect the desired manipulation of the bone fragment.

To calculate the Forward Kinematics required to transform joint coordinates reported by the manipulators 14 into Cartesian coordinates an iterative algorithm is used. Figure 15 is a flow chart illustrating steps taken by the processing system in calculating the Forward Kinematics.

In a first step 200, an initial guess vector \(X_n\) relative to the base 34 \{B\}, containing Cartesian coordinates \(X, Y, Z\) and angles \(\Theta_z, \Theta_y, \Theta_x\), is selected.
In steps 202 and 204 a first estimate \( X_0 \) is set to equal the initial guess vector \( x \) and the Inverse Kinematics are solved based on \( x \) using the equation \( \|i|^2 = R + R p_i - p_i \) set out above to calculate \( q_i \) where \( q_i = \|i| \). The current lengths of the linear actuators 36 termed in this algorithm \( q_{l,m} \), measured by an absolute displacement transducer in step 206, are compared to the calculated lengths (using the Inverse Kinematics) of the linear actuator \( q_i \) and an error is obtained in step 208.

Subtracting \( q_{l,m} \) from \( q_i \) and defining the Forward Kinematics problem as ‘root finding problem’, an error in mm is obtained as mentioned above and compared in step 212 to a suitable and specified tolerance value \( \varepsilon \) in step 210. To find the roots of the error function defined in 208 the Newton-Raphson method is used as a standard algorithm to solve nonlinear equations numerically. Single steps to perform this method are briefly outlined for the present case in 212, 214, 216 and 216.

If \( |f_i(x_n)| < \varepsilon \) is true, i.e. the difference between the calculated length \( q_i \) and measured actuator length \( q_{l,m} \) is smaller than the selected tolerance value \( \varepsilon \), a solution for the Forward Kinematics has been found. In this position the calculated location of the robot platform defined by the Cartesian vector \( X_i \) (where \( i \) donates a single iteration step) corresponds to the measured actuator length \( q_{l,m} \) taking the specified tolerance value \( \varepsilon \) into account.

In case \( |f_i(x_n)| \) is larger than \( \varepsilon \), the Cartesian coordinates summarized by \( X_0 = x_n \) are adjusted using the term \( - f_i(x_n) J(x_n)^{-1} \) specified in step 214 where \( J \) represents the Inverse Jacobian Matrix.

If the absolute value of the adjustment \( \delta X_i \) is less than the specified tolerance value \( \varepsilon \) the correct Cartesian position of the platform 30 in space X has been found based on the joint space input. If not, the calculated adjustment \( \delta X \) in step 214 is added to the initial guess vector \( x_n \) at step 218 and
the steps 202 - 220 are repeated until the correct vector $X$ describing the position and orientation of the platform 30 in Cartesian coordinates is found.

In the embodiment of the system 10 described above with reference to Figure 5, features of the reference bone and bone fragments of a fracture are identified either automatically or by a qualified surgeon based on a graphical representation of the fracture based on the medical images (scans) taken of the fracture prior to and during the surgical procedure for reducing the fracture, and the processing system calculates, from the identified features, manipulations of the reference bone and bone fragments required to effect the desired reduction, and motion reference signals for the manipulations for the controller to cause the first, second and third manipulators 12, 14, 16 to effect the calculated manipulations.

In an alternative second embodiment of the system, the processing system presents a graphical user interface that enables a surgeon to perform a virtual manual pre-reduction on a graphical representation of the fracture. The processing system records manipulations of the virtual bone fragments made by the surgeon, and from these recorded manipulations calculates the motion reference signals required to cause the first, second and third manipulators 12, 14, 16 to effect the manipulations performed on the virtual fragments.

Figure 16 is a screenshot showing a graphical user interface 300 employed in this alternative embodiment, providing a graphical representation of a bone fracture prior to the virtual pre-reduction discussed above.

In the second embodiment to which Figure 16 relates, the steps of segmentation and generation of fragment surface described above in relation to the first embodiment are performed to generate graphical representations 302 of the bone fragments of the fracture which are displayed by the processing system. The image 304 in the right-hand window of the screenshot of Figure 16 shows a graphical representation of the three bone fragments 306, 308, 310 of the fracture case, whilst the left-hand window 312 shows an enlarged view of the fragment 306, illustrating more clearly the calculated fracture surface.

Figure 17 is a screenshot showing a further graphical user interface 320 employed in the second embodiment to permit manipulation of the graphical representations of the bone fragments 306,
308, 310, to achieve a simulated virtual pre-reduction of the fracture. The graphical user interface 320 provides a “virtual joystick” 322 by means of which the surgeon is able to manipulate (translate or rotate) a selected representation of a fragment 306, 308, 310. The effect of such manipulation is shown on a graphical representation 324 of the fracture, allowing the surgeon to see the effect of the manipulations. In this way, the surgeon is able to simulate the manipulations required to each fragment 306, 308, 310 before any manipulation of the actual bone fragments takes place to determine precisely the manipulations required for optimum reduction of the fracture.

The processing system records the manipulations performed on the virtual bone fragments 306, 308, 310, and calculates, from the recorded manipulations, the reference signals which are required to effect the desired manipulations of the actual bone fragments by the first, second and third manipulators 12, 14, 16. These reference signals are transferred to a further graphical user interface, shown at 340 in Figure 18, by means of which the first, second and third manipulators 12, 14, 16 are controlled.

It will be appreciated that the processing and calculation steps will typically be performed by a software program being executed by processing system such as a general purpose computer. Accordingly, the present invention extends to a computer program which, when executed by appropriate processing means, performs processing steps as described above.
CLAIMS

1. A system for reduction of bone fractures, the system comprising:
   a first manipulator for manipulating a bone section of the fracture, the first
   manipulator being attachable to the bone section by means of a percutaneous attachment
   device;
   a second manipulator for manipulating a first bone fragment of the fracture, the
   second manipulator being attachable to the first fragment by means of a percutaneous
   attachment device; and
   a processing system configured to determine reference signals for the first and
   second manipulators required to effect manipulations of the bone section and the first
   fragment required for correct anatomical reduction of the fracture; and
   a controller configured to control the first and second manipulators to cause them
   to perform the manipulations of the bone section and the first bone fragment.

2. A system according to claim 1 wherein the first manipulator is a spatial parallel
   manipulator comprising first and second end sections connected by a plurality of linear
   actuators.

3. A system according to claim 1 or claim 2 further comprising a third manipulator
   for manipulating a second bone fragment of the fracture, the third manipulator being
   attachable to the fragment by means of a percutaneous attachment device, wherein the
   processing system is configured to determine reference signals for the third manipulator
   required to effect manipulations of the second fragment required for correct anatomical
   reduction of the fracture and the controller is further configured to control the third
   manipulator to perform the manipulations of the second bone fragment.

4. A system according to any one of the preceding claims wherein the second and/or
   third manipulator is a spatial parallel manipulator comprising a fixed base and a
   moveable platform connected to the fixed base by a plurality of linear actuators.
5. A system according to claim 4 wherein the fixed base is connected to the moveable platform by six linear actuators.

6. A system according to any one of the preceding claims wherein the controller is configured to cause the manipulators to perform the manipulations of the reference bone and the bone fragment(s) substantially simultaneously.

7. A system according to any one of the preceding claims wherein the processing system is configured to:
   - receive an image of the fracture;
   - segment the image of the fracture to identify fracture surfaces of the reference bone and the bone fragment(s) of the fracture;
   - generate a fragment surface layer for the reference bone and the fragment(s) representative of surfaces of the reference bone and the bone fragments;
   - display a graphical representation of the reference bone and the bone fragments;
   - receive a user input;
   - manipulate the graphical representation to simulate reduction of the fracture based on the user input received;
   - record the manipulations of the graphical representation; and,
   - based on the recorded manipulations, determine the reference signals for the manipulations required for correct anatomical reduction of the fractured bone.

8. A system according to claim 7 wherein the user input is received by means of a virtual joystick presented as part of a graphical user interface by the processing system.

9. A method for calculating manipulations required to effect an anatomical reduction of a bone fracture, the method comprising the steps of:
   - segmenting an image of the fracture to identify fracture surfaces of bone fragments of the fracture;
generating a fragment surface layer for each fragment representative of surfaces of the bone fragment;
calculating an axis of each bone fragment;
calculating fracture surfaces for each bone fragment; and
calculating, based on the axes and fracture surfaces calculated, manipulations required for reduction of the fracture.

10. A method according to claim 9 wherein the segmenting of the image of the fracture comprises calculating Hounsfield intensity values for structures shown in the image and identifying the fracture surfaces from the Hounsfield intensity values.

11. A method according to claim 9 or claim 10 wherein generating the fragment surface layer for each fragment comprises using data from the segmented image to generate a point cloud representing each bone fragment and performing a triangulation on the point cloud.

12. A method according to any one of claims 9 to 11 wherein calculating the axis of a bone fragment comprises defining two sections of the bone fragment, calculating the point centre of each of the defined sections of the fragment and calculating an axis vector for the bone fragment by subtracting the point centre of one section from the point centre of the other section.

13. A method according to claim 12 wherein calculating the point centre of a section of the bone fragment comprises calculating the mean x, y and z coordinates for each point of the point cloud in the section.

14. A method according to any one of claims 9 to 13 wherein calculating the manipulations required for reduction of the fracture comprises calculating a manipulation required for axial alignment of the fragments, calculating an angle of rotation required for
fracture surface alignment and calculating a translation required to close a gap between the bone fragments.

15. A method according to claim 14 wherein calculating the manipulation required for axial alignment of the fragments comprises calculating a transformation required to align the axis vectors of the fragments.

16. A method according to claim 14 wherein calculating the angle of rotation required for fracture surface alignment comprises:
   
   generating a first polyline representative of the fracture surface of a first, reference fragment;
   
   generating a plurality of second polylines representative of the fracture surface of a second fragment, each second polyline having an incremental angular offset with respect to the first polyline;
   
   performing a comparison of the first and each of the plurality of second polylines, wherein the angle of rotation required is determined by calculating the angular offset for which the comparison determines that the first and second polylines are most similar.

17. A method according to claim 16 wherein performing the comparison comprises:
   
   calculating a first cross correlation coefficient for the first polyline;
   
   for each of the plurality of second polylines, calculating a second cross correlation coefficient; and
   
   calculating a difference value between the first and second cross correlation coefficients, wherein the angle of rotation required is the angular offset of the second polyline for which the difference value is smallest.

18. A method for calculating manipulations required to effect an anatomical reduction of a bone fracture, the method comprising the steps of:
segmenting an image of the fracture to identify fracture surfaces of bone fragments of the fracture;

generating a fragment surface layer for each fragment representative of surfaces of the bone fragment;

displaying a graphical representation of the bone fragments;

receiving a user input;

manipulating the graphical representation of one of the bone fragments to simulate reduction of the fracture based on the user input received;

recording the manipulations of the graphical representation of the bone fragment;

and,

based on the recorded manipulations, determining reference signals for manipulations of the fragments of fractured bone required for correct anatomical reduction of the fractured bone.

19. A method according to claim 18 wherein the user input is received by means of a virtual joystick presented as part of a graphical user interface.

20. A system according to any one of claims 1 to 6 wherein the processing means is configured to perform the method of any one of claims 9 to 19, and the controller is configured to cause the manipulators to perform the calculated manipulations.

21. A computer program which, when executed on appropriate processing means, performs the method of any one of claim 9 to 19.
ABSTRACT

A SYSTEM FOR ANATOMICAL REDUCTION OF BONE FRACTURES

The present application relates to a system (10) for anatomical reduction of bone fractures in which first and second manipulators (12, 14), and optionally a third manipulator (16), are attached to fragments of the fracture to be reduced by percutaneous attachment devices such as Schanz pins. An underlying processing system determines, from one or more medical images of the fracture, manipulations such as rotations and translations of the bone fragments required to correctly reposition and align the fragments for optimum healing of the fracture. The processing system provides motion reference signals (position, speed, acceleration) for a controller, which in turn causes the first, second and third manipulators (12, 14, 16) to effect the calculated manipulations.

(Figure 1)
Start

Pre-operative CT scan in the hospital (diagnoses)

Start of fracture surgery

Insertion of Schanz pins in each fragment by the surgeon

Fixation of a reference bone to ER followed by an intra-operative CT scan

Registration of 3D image data and cartesian motion data

Manual or automatic extraction of bone features (surfaces, points etc.) for automatic alignment

Features identified

Coarse and fine registration of fragments based on extracted features

Transformation parameters (T and R) received

Automatic physical reduction of fragments via inserted Schanz pins and IR

Fragments reduced

Intra-operative verification CT scan I

False

Appropriate reduction accuracy?

True

Manual fixation of fragments by the surgeon

Intra-operative verification CT scan II

False

Appropriate reduction accuracy?

True

End of semi-automatic and minimally invasive fracture reduction

End
Define $\theta_{\text{max}}$ and $\dot{\theta}_{\text{max}}$

Calculate IK to solve $s_{eq} = \left| \dot{s}_{eq} \right|$

Calculate the parameters $(\dot{l}_{eq}, \ddot{l}_{eq})$ of the velocity profile

Determine the leading robot axis $l_{eq}$

Yes

if $l_{eq} = l_{eq}$

No

Adjust the velocities of the remaining robot axes $\theta_{\text{adj}}$

$$\dot{\theta}_{\text{adj}} = \frac{\ddot{\theta}_{\text{max}} l_{eq}}{2} - \frac{\ddot{\theta}_{\text{max}} l_{eq}^2}{4} - (s_{eq} / \dot{\theta}_{\text{max}})$$

Calculation of path data $(s(t), \dot{s}(t), \ddot{s}(t))$ for the acceleration phase $(t \leq t_{eq})$, for the phase of constant velocity $(t_{eq} < t \leq t)$ and the deceleration phase $(t \geq t)$ of the velocity profile

for acceleration phase

$s(t) = \theta_{\text{max}}$  
$s(t) = \dot{\theta}_{\text{max}} t$  
$s(t) = 1/2 \dot{\theta}_{\text{max}} t^2$

for phase of constant velocity

$s(t) = 0$  
$s(t) = \theta_{\text{adj}}$  
$s(t) = (1/2 \dot{\theta}_{\text{adj}}) / \dot{\theta}_{\text{max}}$

for deceleration phase

$s(t) = \theta_{\text{max}} - \dot{\theta}_{\text{max}} (t - t_{eq})$  
$s(t) = \theta_{\text{adj}} - (t_{eq} - t) / \dot{\theta}_{\text{max}}$  
$s(t) = \theta_{\text{adj}} - t (1/2 \dot{\theta}_{\text{adj}}) / \dot{\theta}_{\text{max}}$

Calculation of joint parameters $\dot{q}(t), \ddot{q}(t), \dddot{q}(t)$ using path data and sign function

$y = \text{sign}(x) = \begin{cases} -1, & \text{for } x < 0 \\ 0, & \text{for } x = 0 \\ 1, & \text{for } x > 0 \end{cases}$
Chose initial guess vector 
\[ X_n = (x, y, z, \theta_x, \theta_y, \theta_z) \] 
relative \{B\}

\[ X_0 = x_0 \]

Solve IK based on \( x_0 \)

Measure current leg lengths from position transducer \( q_{i,m} \)

Calculate 
\[ e = f_i(x_n) = q_i^2 - q_{i,m}^2 \]

Specify a tolerance value \( \varepsilon \) and

\[ |f_i(x_n)| < \varepsilon \]

YES

\[ \partial X_i = \left[ -\frac{f_i(x_n)}{J(x_n)^{-1}} \right] \quad i = 1, 2, \ldots, 6 \]

NO

Adjustment

\[ |\partial X_i| < \varepsilon \]

YES

Solution 
\[ X_n = (x, y, z, \theta_x, \theta_y, \theta_z) \]

\[ x_n = x_n + \partial X \]
Figure 17
Figure 18